Stakeholder Collaboration in Air Force Acquisition: Adaptive Design Using System Representations

by **Robert E. Dare**

Bachelor of Science, Aeronautics and Astronautics
Massachusetts Institute of Technology, 1984
Master of Science in Engineering, Option in Astronautical Engineering
West Coast University, 1991

Submitted to the Engineering Systems Division
in Partial Fulfillment of the Requirements for the Degree of
DOCTOR OF PHILOSOPHY
in TECHNOLOGY, MANAGEMENT AND POLICY
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2003

Signature of author…………………………………………………………………………………
Technology, Management and Policy Program
2 May 2003

Certified by…………………………………………………………………………………
Earll M. Murman
Ford Professor of Engineering
Department of Aeronautics and Astronautics and Engineering Systems
Thesis Supervisor and Thesis Committee Chairman

Certified by…………………………………………………………………………………
Sheila E. Widnall
Institute Professor and Professor of Aeronautics and Astronautics
Thesis Committee Member

Certified by…………………………………………………………………………………
Thomas J. Allen
Howard W. Johnson Professor, Sloan School of Management
Thesis Committee Member

Certified by…………………………………………………………………………………
Eric S. Rebentisch
Research Associate, Center for Technology, Policy and Industrial Development
Thesis Committee Member

Accepted by…………………………………………………………………………………
Daniel E. Hastings
Professor of Aeronautics and Astronautics and Engineering Systems
Director, Technology & Policy Program
Co-Director, Engineering Systems Division
Chair, Committee on Graduate Studies

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **00 JUN 2003**	2. REPORT TYPE **N/A**	3. DATES COVERED **-**

4. TITLE AND SUBTITLE **Stakeholder Collaboration in Air Force Acquisition: Adaptive Design Using System Representations**	5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Massachusetts Institute of Technology**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S) 11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES
The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**UU**	**333**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

(This page intentionally left blank)

Stakeholder Collaboration in Air Force Acquisition: Adaptive Design Using System Representations

by

Robert E. Dare

Submitted to the Engineering Systems Division on May 2, 2003 in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Technology, Management, and Policy

Abstract

Air Force development of new or evolutionary weapon systems is a complex endeavor due to the involvement of many stakeholders and the presence of considerable uncertainty in the acquisition environment. The ability to adapt a weapon system while it is still being designed affords a means to respond to this complexity. The fundamental motivation for this research is to discover how Air Force development programs, operating within established constraints, can improve their adaptability during the design phase to provide more value to the warfighter.

The thesis of this research is that the quality and nature of collaboration between stakeholders during the design phase of weapon system development programs determines how effectively they share knowledge, which in turn drives the level of program adaptability.

Eight case studies were conducted on Air Force development programs. Data were collected on collaborative practices and patterns of adaptability demonstrated during design. The research placed an emphasis on usage of "system representations" such as prototypes and beta software releases that acted as a form of boundary object to facilitate knowledge sharing across organizational boundaries.

As programs used system representations to provide higher levels of knowledge sharing, they were found to be more adaptable. System representations were more effective at promoting adaptability when they represented the design with higher fidelity, providing system-level detail and covering stakeholder emphasis areas. Lastly, certain key stakeholder roles were found to contribute both flexibility and structure, facilitating a "zone of novelty" in which the stakeholders could exercise creativity and evaluate design options while still executing the program within established constraints.

This research indicates that the pressing need for Air Force programs to be able to adapt in today's uncertain acquisition environment can be addressed to a significant degree through the usage of effective system representations in conjunction with supporting patterns of stakeholder interaction. Specific recommendations for Air Force acquisition policy makers and practitioners are provided.

Thesis Advisor:
Earll M. Murman
Ford Professor of Engineering

Department of Aeronautics and Astronautics and Engineering Systems

Committee:

Sheila E. Widnall
Institute Professor and Professor of Aeronautics and Astronautics

Thomas J. Allen
Howard W. Johnson Professor, Sloan School of Management

Eric S. Rebentisch
Research Associate, Center for Technology, Policy and Industrial Development

Stakeholder Collaboration in Air Force Acquisition: Adaptive Design Using System Representations

Robert E. Dare
Lean Aerospace Initiative
Massachusetts Institute of Technology

Executive Summary

This research, conducted under the auspices of the Lean Aerospace Initiative, sought to determine how Air Force development programs could achieve high levels of adaptability during the design phase of acquisition while maintaining effective management of program risk. Due to tremendous uncertainty faced by many development programs in such areas as requirements, technology and funding, traditional planning and measurement efforts, with their emphasis on stability, must be complemented by efforts to promote adaptability. The thesis of this research is that the quality and nature of collaboration between stakeholders during the design phase of weapon system development programs determines how effectively they share knowledge, which in turn drives the level of program adaptability.

To gain insight into the phenomena of stakeholder collaboration and adaptability, this research undertook retrospective studies of eight development programs, focusing on the design phase. During design, changes are typically more affordable than they are during the ensuing test period because they involve less rework. Command and Control (C2) systems were selected for the eight studies because of their acute need to manage change over time, which arises from the rapid rate of technology change in the areas of communications and computers.

The research focused on collaboration between three major stakeholders who contribute unique knowledge and fill different roles during the design phase. The first stakeholder is the user community or warfighter, consisting of organizations that will be the eventual operators of the system. Second is a government acquisition agency, or System Program Office (SPO), whose role is to establish and oversee one or more contracts with private industry to perform the development work within established programmatic constraints. The third major stakeholder is a prime contractor who develops the system in accordance with the government contract. Interviews with knowledgeable representatives of the three stakeholders were combined with in-depth reviews of program documentation to reconstruct the collaborative patterns and adaptive results of each program.

The case studies centered on two aspects of collaboration. The first was a specific collaborative mechanism called a "system representation" (SR), such as a prototype or beta software release, which was used as a means to share partial information about an ongoing system design. Unlike briefing slides or documents, a SR allows a stakeholder to visualize and interact with the system as it is envisioned at that point in the design process. The second, related aspect of collaboration involved stakeholder roles that enhanced adaptability while maintaining an acceptable level of program risk.

The primary research questions undertaken in this study were the following.

- How does a system representation enhance adaptability?
- What characteristics make system representations effective at promoting adaptability?
- What are the roles of stakeholders in facilitating program adaptability?

- Do certain characteristics of programs (requirements uncertainty, funding level and duration of design phase) predispose them to be more or less adaptable?

Table 1 provides highlights of the programs (A through H) that were studied.

	Program A	**Program B**	**Program C**	**Program D**
Adaptive strengths	-User & contractor co-located -User technical expertise -SR used operationally	-SR at SPO and contractor facilities -Open communication - Shared objectives	-Prototype gave baseline for req'ts & design -Built up-front consensus on design	-SR helped resolve issues -Changes anticipated and welcomed
Adaptive weaknesses	None	-4 users -Lacked stable concept of ops.	-No SR access -Remote users -$ constrained -No formal concept of ops.	-Req'ts not well developed -$ constrained -4 users
R&D ($ mil.)	24	23	13	22 (appx.)
Design (mos.)	43	15	14	16
System Representation	Development system	Development software	Fielded prototype	Development software
Adaptability	Very high	Very high	Moderate	Moderate

	Program E	**Program F**	**Program G**	**Program H**
Adaptive strengths	-Informal user feedback -Contractor made changes informally	- SPO in touch with legacy system users	-On-site testers had ops experience -Planning for tech. insertion, changes -Life cycle cost decision model -Early planning	-Detailed issue discussions -Informal user feedback -Strong SPO systems engineering
Adaptive weaknesses	-$ constrained -Low priority -SPO and user friction -User HQ lack of operational experience	-Two SPOs -Long decision process (early) -No concept of operations -Job scope underestimated -Limited user involvement	-Field user busy	-Limited user interaction with contractor -Late req'ts -Late radio -Program highly constrained/complex -Job scope underestimated -No formal concept of ops.
R&D ($ mil.)	15 (appx.)	28 (appx.)	140	40
Design (mos.)	6	21	24	24
System Representation	Development software	Development software	Representative lab	Representative lab
Adaptability	Moderate	Low	High	High

Table 1. Summary of Programs A through H

Analysis of case study data established adaptability levels achieved by each program (ranging from "very high" to "low") and led to a set of findings and recommendations regarding system representations and stakeholder roles.

The first finding regarding system representations (SR) was that **adaptive programs used a SR to share knowledge between stakeholders.** Figure 1 summarizes the data that led to this finding. Programs that experienced more extensive knowledge sharing due to SR usage tended to have higher levels of program adaptability.

Knowledge Sharing with SR / ***Adaptability***	**Exceptional**	**Strong**	**Moderate**	**Weak**	**None**
Very high	**A**		**B**		
High		**G**		**H**	
Moderate			**D**	**E**	**C**
Low				**F**	

DEFINITION OF CRITERIA (SUMMARY)
-Exceptional – in depth stakeholder interaction (fully exercising SR) on a daily basis
-Strong –substantive interaction (exercising some aspects of SR) on a daily basis
-Moderate – substantive interaction (exercising some aspects of SR) on a weekly basis or less
-Weak – top-level interaction (brief exposure to some aspects of SR) on a weekly basis or less
-None – no interaction

Figure 1. Knowledge sharing with SR versus adaptability

The second finding related to system representations was that **more adaptable programs had higher fidelity system representations (system level detail and coverage of stakeholder emphasis areas).** Figure 2 presents the data related to the first SR fidelity measure - level of detail (system, subsystem or minimal).

Level of Representation / ***Adaptability***	**System Level Detail**	**Subsystem Level Detail**	**Minimal Design Detail**
Very High	**A, B**		
High	**G**	**H**	
Moderate			**C**
Low			**F**

DEFINITION OF CRITERIA
- System level: portrays overall system functionality and interaction of subsystems
- Subsystem level: portrays subsystem functionality
- Minimal: minimal representation of functionality

Figure 2. Detail level of SR versus adaptability

Stakeholders for each program indicated that they had emphasis areas for their programs. Figure 3 consolidates the data related to the second measure of SR fidelity, level of coverage of emphasis areas. Programs had either high or low SR coverage.

	High SR Coverage	Low SR Coverage
Very High to High Adaptability	**A, B, G**	**H**
Moderate to Low Adaptability	**E**	**C, D, F**

OBSERVED GOVERNMENT EMPHASIS AREAS:

- *Technical performance*
- *User interface*
- *Interoperability*
- *Maintenance*
- *Life cycle cost*
- *Reliability*
- *Development cost*

In highly adaptive programs, stakeholders were observed to perform roles that contributed in significant ways to essential adaptability functions. Table 2 consolidates the roles that were observed to be best practices supporting these adaptability functions.

Adaptability Function	**SPO Role**	**User Role**	**Contractor Role**
Demonstrate partial design	• **Encourage and facilitate user engagement** • **Manage user expectations**	• **Coordinate field participation**	• **Create and share SR**
Identify potential design or requirements changes		• **Provide design feedback: operational perspective (how system will be used)**	
Evaluate potential changes	• **Facilitate contractor evaluation** • **Evaluate risks**	• **Define priorities (importance of potential changes)**	• **Evaluate cost, benefit and best implementation approach**

Table 2. Key stakeholder roles supporting program adaptability functions

This set of stakeholder roles provided a mixture of flexibility and structure for the most adaptable programs, encouraging both innovation and risk management.

The final research question dealt with the relationship between program characteristics and adaptability. Data indicated that the program characteristics of requirements uncertainty, research and development (R&D) budget, and design phase duration, were not correlated with levels of adaptability.

This research resulted in the following recommendations.

1. Make system representations (SR) and adaptability part of acquisition planning
    - Evaluate benefit vs. cost of SR during pre-contract planning
    - Plan resources
        - SR creation and modification
        - Evaluation of potential changes
        - Implementation of changes

2. Involve the operational user in the design phase
    - Provide feedback on design based on unique knowledge of operational considerations
    - Provide integrated, up to date priorities

    (Note: a system representation gives user representatives a mechanism to be productively engaged in design)
3. Create effective system representations
    - Provide system level detail
    - Provide coverage of stakeholder emphasis areas
    - SRs are strongest at portraying visual emphasis areas: technical performance (functionality), user interface and maintenance
    - Analysis (computational assessments) helps with coverage of reliability, development cost and life cycle cost
    - SRs and analysis are effective as complements
4. Make effective use of system representations
    - Once the contractor can make functionality visible, there may be value in sharing a SR
    - Make aspects of the design visible and provide opportunity for interaction
    - More in-depth interaction and greater frequency leads to greater knowledge sharing (balance against resource considerations)

5. Create a "zone of novelty" – a mix of flexibility and structure for the program
    - Exercise key stakeholder roles (Table 2) that support adaptabilityThis research

also applied two theoretical lenses related to inter-organizational interaction and

adaptability: complex adaptive systems theory (CAS) and boundary objects. CAS organizational constructs were developed, representing considerations that organizations should address to promote adaptability. Findings correlated well with these constructs, implying they may have application in other inter-organizational settings.

CAS constructs:

- Develop tools and procedures for information sharing
- Look for and resolve potential perturbations to stability
- Balance structure and flexibility

Several authors, including S. L. Star, Paul Carlile and Josh Bernstein, have explored the concept of using a boundary object to facilitate knowledge transfer across knowledge boundaries. This research expanded considerations of boundary objects to inter-organizational settings and defined a special type of boundary object called a system representation (SR). Data indicated that SRs were effective boundary objects, helping stakeholders bridge knowledge boundaries to establish shared understanding.

This research has provided recommendations for acquisition policy makers and stakeholders on how programs can be made more adaptable. It has provided evidence that CAS principles are important considerations to understand inter-organizational interactions and adaptability. The research has also expanded application of the concept of a boundary object to include inter-organizational contexts. Finally, the research has clarified the importance of adaptability during design. The definition of value for the warfighter changes over time, and there are limited resources available for modification of fielded systems. Therefore, the design phase is a unique time when it is possible to visualize and interact with the system when changes are still affordable.

Acknowledgements

I want to recognize a number of individuals who helped make this dissertation possible. I start with my committee, who provided exceptional intellect, vision, moral support and a marvelously diverse set of skills and perspectives. I thank Prof. Earll Murman, my committee chair, for somehow knowing exactly what I needed and what to say to guide me through this journey. Earll's wisdom and compassion were invaluable. I am grateful to Prof. Sheila Widnall for sharing her exceptional insight, observations and ideas for emphasis and improvement. I thank Prof. Tom Allen and Dr. Eric Rebentisch for providing rigor and depth of thought to ensure this work would represent a meaningful contribution. In addition to individual contributions, the committee achieved a synergy during this effort that was almost miraculous.

I am also grateful for the major contributions of many other MIT faculty members, including Daniel Hastings, Paul Carlile, Dave Mindel, Richard de Neufville, Bill Lucas, John Carroll, Starling Hunter, Debbie Nightingale, Col. John Keesee and Col. Pete Young. Others at MIT who helped in times of need included Peggy Udden, Fran Marrone, and Fred Donovan.

I hold the Technology and Policy Program (TPP) and the Technology, Management and Policy doctoral program in high regard and will always cherish my affiliation with these academic entities. I want to say a special thanks to Gail Hickey, Liz Zotos, Sydney Miller, Jean Marie DeJordy, and the rest of the TPP/TMP staff – you demonstrated a caring and professionalism that made a significant difference in the lives of the students in these programs.

I also received great support from Col John Kuconis, Ann Cronin, and the exceptional officer and enlisted corps of Det. 365 AFROTC at MIT. The cadre is an important part of building the future of the Air Force, and I am proud to have had a chance to work with them. I owe a thank you to my fellow AFIT officers at MIT, including Maj. Dan Caputo, Capt. Dave Ferris, Capt. Brandon Wood, Capt. Clark Allred, Capt. Brett Conner, Capt. Ben Brandt and many others.

I wish to thank the Lean Aerospace Initiative (LAI) community of faculty, staff, students and partners from industry, labor and government. I am especially grateful for the intellectual contributions of Kirk Bozdogan, Joyce Warmkessel, Tom Shields and Hugh McManus.

I also benefited greatly from discussions of many kinds with my fellow graduate students. The list includes (with apologies for lapses in memory): Josh Bernstein, Angie Kelic, Jay Falker, Larry Seigel, Alexis Stanke, Jim Chase, Cory Hallam, Annalisa Weigel, Myles Walton, Jeffrey Munson, Jacob Markish, Major Chris Forseth, and Brian Ippolito. Also, those of us in TPP 99 will always share a special bond – thanks for many memorable discussions and experiences!

Maj. Ross McNutt was a unique contributor whose advice steered me back to MIT. As Ross generously shared his experiences, both before and during this endeavor, he helped me keep my sanity and perspective. Ross's entrepreneurial spirit has been an inspiration for this effort.

I am also indebted to the many individuals in Air Force System Program Offices (SPOs), in the user community, and in industry who gave of their valuable time to share their insights with me during the course of this research. These individuals remain unnamed to protect their confidentiality. Unquestionably, this effort would not have been possible without their candor and understanding of the issues at hand. Several individuals at the Air Force's Electronic Systems Center (ESC) were also instrumental in assisting this research. These included Col. Joseph Maryeski, Col. David Chaffee, Col. Bud Vazquez, Col. Randy Smith, Mr. Rich Bleu, Col. Chris King, Ms. Pauline Froebel, Lt. Rob Mishev, Lt. Alan Driver, Ms. Sherin Shami-Campbell, and Ms. Margaret Cefalo.

I am thankful for the vision of senior leadership in the Air Force as they support the role of higher education in development of the officer corps. I appreciate the support of Captain Rick Sutter, Major Melissa Flattery and Lieutenant Angela Bjorge of the Air Force Institute of Technology (AFIT). These individuals helped make my AFIT experience a very positive one.

Most importantly of all, I want to thank my wife Fawn, and my son, Jonathan for putting up with all the hours I was locked away with my computer and my books, and for being constant reminders of what is most important in life. Thank you for being there for me, and for the many sacrifices you made in the name of our future together. This effort would mean nothing without you.

The author gratefully acknowledges the financial support for this research from the Lean Aerospace Initiative, a unique research consortium with membership from the U.S. government, academia, industry and labor.

I dedicate this thesis to my mother, Margery Ann Wade, and to my son, Jonathan Quinton Dare.

The ongoing cycle of life sustains us through remembrance of the past and hope for the future.

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

1.1 Motivation

Air Force development of new or evolutionary weapon systems is a complex endeavor, in part because of the involvement of many stakeholders with differing and time-varying needs and constraints. Complexity also stems from an uncertain environment that includes shifting budgets, evolving threats and rapid technology advances. Because of these considerations, emphasis on up-front planning is necessary but not sufficient. Ongoing interaction among stakeholders provides a means of establishing a shared understanding of program issues and opportunities as the development program unfolds. The ability to adapt a program affords a means to respond to complexity, providing a way to address unforeseen challenges and potentially add value for the warfighter beyond what was envisioned in original program plans. The thesis of this research is that the quality and nature of collaboration between stakeholders during the design phase of weapon system development programs determines how effectively they share knowledge, which in turn drives the level of program adaptability.

In a recent Air Force acquisition policy memo (Sambur, 2002), the Air Force chief acquisition officer stated, "the primary mission of our acquisition system is to rapidly deliver to the warfighters affordable, sustainable capability that meets their expectations." The memo expresses that "success hinges on up-front, collaborative and concurrent planning" of stakeholders to "establish, at the outset of the program, mutual, realistic expectations for content delivered, schedule of delivery, and cost." The policy also advocates Evolutionary Acquisition (EA) as the "preferred acquisition strategy" and

spiral development as “the preferred process to execute the EA strategy.” EA and spiral development, as discussed in Chapter 2, provide means to re-evaluate plans periodically, but in-depth mechanisms for making these approaches work, including revised definitions of the roles of the stakeholders, are not fully understood. This research endeavors to contribute some answers to the dilemma of how to accommodate dynamic considerations from different stakeholders and from external factors, and achieve the greatest value for the warfighter within established program constraints. Ongoing stakeholder collaboration is the first step in addressing this challenge.

In order to gain insight into the phenomena of stakeholder collaboration, this research undertook retrospective studies of eight development programs, focusing on the design phase. During design, changes are typically more affordable than they are during the ensuing test period because they involve less rework. The design phase is considered for this study to be the period after definition of formal requirements and through the completion of design definition, which typically happens at a Critical Design Review (CDR) or equivalent milestone. Command and Control (C2) systems were selected for the eight studies because of their acute need to manage change over time, which arises from the rapid rate of technology change in the areas of communications and computers.

The research focused on collaboration between three major stakeholders who contribute unique knowledge and fill different roles during the design phase. The first stakeholder is the user community or warfighter, consisting of organizations that will be the eventual operators of the system. Typically an Air Force major command headquarters office will be responsible for representing this community by tracking and communicating user needs in the form of program requirements. Second is a government

acquisition agency, or System Program Office (SPO), whose role is to establish and oversee one or more contracts with private industry to perform the development work within established programmatic constraints. The third major stakeholder is a prime contractor who develops the system in accordance with the government contract. Other stakeholders, including most notably the government oversight community and the test community also play a role in the design phase. However, it was necessary to focus on three key participants to construct an executable research regime. Interviews with knowledgeable representatives of the three stakeholders were combined with in-depth reviews of program documentation to reconstruct the collaborative patterns and adaptive results of each program.

The case studies centered on a specific collaborative mechanism that was identified during earlier exploratory research. Some programs were found to use a "system representation" (SR), such as a prototype or beta software release, as a means to share partial information about an ongoing system design. A system representation (SR) is defined in this research as a visible, interactive representation of the contractor's system design as it is envisioned at a point in time. Of the eight programs studied for this research, all adapted to widely varying degrees, but programs that created and used a SR found that it had a positive impact on the ability of stakeholders to share knowledge, enabling them to develop a shared understanding about the system. SRs made the design accessible to all stakeholders and aided in identification and evaluation of potential adaptations. High fidelity SRs, which captured system-level design details and covered stakeholder emphasis areas, were found to be effective in promoting adaptability. Also, certain stakeholder roles were found to be critical in establishing a balance between

structure and flexibility that permitted stakeholders to adapt while maintaining an acceptable level of program risk.

Stakeholders in these eight development programs exchanged different types of knowledge, including evolving user needs, new technology options, operational implications of design choices, user priorities, and programmatic constraints. The wide exchange of relevant and accurate information seemed to be key to identifying and evaluating potential changes that could add value to the program. The role of the user community was particularly valuable in some of the programs in evaluating the evolving design to provide timely feedback on operational considerations and to prioritize potential changes in the requirements or design.

Insights from the eight cases implied that the concepts of collaboration and adaptability are interrelated in the context of system design. Schrage defines collaboration as “shared creation and/or shared discovery.” (Schrage, 1995). Generating a system design is a creative process, but it is also a discovery process in which initial expectations are often subject to change. As the prime contractors that were studied created a system design to fulfill program requirements, design information was often shared with the SPO and the user community in some fashion. Stakeholder interaction, in particular regarding requirements clarification and exploration of design trade spaces, resulted in design being a shared creation process. In several of the cases, a SR helped make this interaction more substantive. In the context of this research, collaboration refers to sharing knowledge between stakeholders about the system during design with the intent to identify and disposition emergent issues and opportunities.

For purposes of this study, adapting refers to a decision to change a program requirement or to modify a currently envisioned design choice as a result of stakeholder collaboration. Sharing knowledge through collaboration enables stakeholders to reach consensus and make an informed, mutual decision to approve a change. Since Air Force development programs are highly constrained, adapting may involve the risk of violating these constraints. Taking managed risks provides the chance to add value for the warfighter, as might happen when a program incorporates a new technology or implements a solution to a parts-obsolescence issue. Stakeholder interaction can provide a significant means of managing risk, to the extent that it allows decision-makers to know the relevant factors impacting a decision to change. In the cases that were studied, these factors included the benefit, priority, cost and risk associated with a potential change.

It is important to draw a contrast between adapting and a practice known historically as "gold plating." The practice of providing the warfighter more capability without regard to funding constraints, or gold plating, has historically been a major cause of cost growth for some development programs. However, there are several legitimate scenarios for adaptation that factor in cost risk considerations. Sometimes changing a requirement or design choice has no cost or saves money, especially if the decision is made early in the program. If an adaptation will incur minimal costs, the resources may be available in management financial reserves. An adaptation may have enough priority to displace or modify an existing requirement, offsetting any cost increase. In other cases, a change may add so much utility that additional funding can be justified to higher authorities. Adapting makes sense in the proper context, and failure to adapt due to risk aversion leads to lost opportunities.

The primary research questions undertaken in this study were the following.

- How does a system representation enhance adaptability?
- What characteristics make system representations effective at promoting adaptability?
- What are the roles of stakeholders in facilitating program adaptability?
- Do certain characteristics of programs (requirements uncertainty, funding level and duration of design phase) predispose them to be more or less adaptable?

1.2 Relationship to Lean Principles

During the development of the area of focus for this research, many of the recurring themes had their origins within the Lean Aerospace Initiative (LAI), a research consortium at the Massachusetts Institute of Technology. In particular, LAI emphasis on delivery of best value weapon systems to the warfighter has been a central consideration. The need to define value for a program leads to another important LAI theme -- the recognition of multiple stakeholders whose needs and contributions can be combined to make up a value proposition for weapon systems. LAI research (Stanke, 2001) has found evidence that successful programs formulated and maintained a value proposition between key stakeholders. The imperative to respond to evolving stakeholder needs in an uncertain environment has led to a focus for this research on the theme of adaptability as an enabler of best value delivery.

LAI has developed a Lean Enterprise Model (LEM) as "a systematic framework for organizing and disseminating research results…designed to help LAI members

identify and assess the leanness of their own organizations and processes." (LAI, 1998). Five of the LEM's twelve overarching practices have a relationship to this research:

- Assure seamless information flow – an enabler of effective collaboration.
- Make decisions at lowest possible level – effective, rapid decisions enhance adaptability.
- Develop relationships based on mutual trust and commitment – also central to effective collaboration.
- Continuously focus on the customer – provides "true north" definition of what potential adaptations represent added value.
- Nurture a learning environment – collaboration promotes learning about different aspects of the system under development, including stakeholder values and constraints.

One of the two meta-principles cited in the LEM is "responsiveness to change", which is analogous to adaptability. One of the four LEM enterprise principles is "effective relationships within the value stream", which implies collaboration. Therefore, the concepts of adaptability and an emphasis on the importance of collaboration are integral to the lean principles espoused by LAI.

Another major LAI theme has been the necessity to both "do the job right" and "do the right job" (Murman, et al, 2002). Doing the job right involves procedures. Doing the right job requires collaboration between stakeholders to refine a shared definition of the job over time.

While the original focus of lean principles was on production, LAI has played a major role in expanding the application of lean concepts to such areas as product

development and acquisition. The latest phase of LAI has advocated and provided support for a further expansion of lean principles to the enterprise level (Murman, et al, 2002.) This perspective is in keeping with current Air Force policy to enhance collaboration between stakeholders and deliver systems faster and with greater credibility (Sambur, 2002.) As of this writing, the Air Force is engaged with LAI in a major initiative entitled "Lean Now!" which seeks to expand the application of lean principles to Air Force programs and processes. As the Air Force seeks greater flexibility and timeliness in the acquisition of systems, the concepts of collaboration and adaptability are at the forefront of current lean thinking.

1.3 Dissertation Overview

Chapter 2 provides an explanation of the specific context of Air Force acquisition. This chapter includes a summary of regulations that describe the design process and define the roles of the major stakeholders. It covers recent Air Force policy related to relevant acquisition themes such as stakeholder interaction, adaptability and responsiveness to warfighter needs. Also, the chapter reviews defense reform literature for insight into the acquisition environment, tensions between stakeholders, and the importance of collaboration and adaptability.

Chapter 3 reviews literature associated with the theoretical underpinnings of the research: systems theory, complex adaptive systems (CAS) theory and the concept of boundary objects. These bodies of thought lead to two observations. Development programs adapt because of stakeholder interactions across organizational boundaries.

Also, boundary objects are important enablers to adaptability because they allow Air Force and industry stakeholders, who have diverse perspectives and knowledge, to develop shared understanding during system design. This discussion, in combination with the Air Force context covered in Chapter 2, leads to the development of four research questions that guided the research design.

Chapter 4 covers the research method that was used to collect data, including the sampling strategy and data collection process. It also defines a set of variables, discusses the relevance of stakeholder roles and explores validity considerations of the research design.

The case study data is summarized in Chapter 5, which describes the 8 cases individually. Each case write-up addresses program context, stakeholder roles, description and usage patterns for system representations, stakeholder interactions and program adaptability. A summary matrix is provided at the end of the chapter.

Chapter 6 initiates analysis of the case study data, covering the thought process and data used to differentiate between programs based on their adaptability, how a SR helps collaboration and adaptability, and what characteristics of SRs make them more effective at promoting adaptable results. Relationship of these results to theory is also described.

Chapter 7 continues the data analysis, examining the significance of stakeholder roles as they relate to levels of program adaptability. The functions associated with adaptability are broken down into the key stakeholder roles observed in highly adaptive programs. Patterns in these roles are explored using concepts from complex adaptive systems theory.

Chapter 8 presents a summary of findings and provides recommendations and discussion related to the findings. The chapter also addresses promising avenues for further research.

Chapter 2 The Air Force Acquisition Context

2.1 Introduction

The purpose of this chapter is to provide a description of Air Force elements that are central to this research including the primary stakeholders who are involved, the design phase of development, and current acquisition policies impacting collaboration and adaptability. To supply this context, information is drawn from three sources: the body of regulations and guidance provided by the Department of Defense (DoD) and the Air Force; defense reform literature; and the author's exploratory research on Air Force stakeholder collaboration.

Regulations and guidance provide information regarding the primary stakeholders, the design phase and current policy issues. The defense reform literature describes the defense acquisition environment and provides insight into stakeholders, including their interactions and incentives. Finally, the author's exploratory research contributes information regarding barriers and enablers to collaboration and mechanisms for collaboration during development.

2.2 Insight from Government Regulations and Guidance

2.2.1 Stakeholders in AF Acquisition

Arguably the most important stakeholder during acquisition is the operational user who will engage in future conflicts with the system that is under development. As described below, the user community is represented by a lead command during the

identification of requirements for a new or modified system. However, the Air Force depends on the acquisition expertise of Air Force Materiel Command (AFMC) personnel to oversee the translation of these requirements into an operational system.

AFMC's organizational unit for managing weapon system acquisitions is the System Program Office (SPO.) The SPO is a central stakeholder, providing the bridge between the user community and the actual developers in industry.

The third pivotal stakeholder is the industry prime contractor, who enters into a contractual relationship with the SPO to perform the development work. Specifics of the prime contractor role are spelled out in the contract.

Other stakeholders include government oversight agencies in DoD and the Air Force, and test and evaluation, budget and sustainment organizations, as well as others. The role played by these additional stakeholders in the design of new weapon systems can be significant. There were two primary reasons for limiting the research to collaboration between the SPO, user and contractor. The first consideration was the practical need to bound the data collection process. The second was the observation that these three stakeholders were the key players influencing the design process. Users provide the input (requirements) to design, the contractor provides the output (the design itself), and the SPO functions in the middle as an integrator and enforcer of rules, constraints and best practices.

The following descriptions draw from Air Force regulations, instructions and policy directives to define the roles of the lead command, the SPO and the prime contractor. Understanding the roles of the stakeholders is an essential prerequisite to studying stakeholder collaboration. An argument is presented that each of the

stakeholders, because of their role, possess unique knowledge that pertains to the design process. The existence of unique, relevant knowledge distributed among stakeholders provides a key rationale for the importance of stakeholder collaboration in support of quality decision-making and provides some of the logical underpinnings for this study's research questions.

2.2.1.1 Lead Command (Government User)

The role of the lead command is specified in Air Force Policy Directive 10-9, "Lead Operating Command Weapon Systems Management", referred to as AFPD 10-9 (2000). This document specifies that, "The Air Force assigns responsibility for overall management of each system to a "lead command" to ensure that all requirements associated with every system receive comprehensive and equitable consideration." The lead command is "the overall advocate for that system over its life cycle." Per AFPD 10-9, responsibilities of the lead command include:

- Planning, programming and budgeting for designated system-wide unique equipment, modifications, initials spares, replenishment spares, and follow-on test and evaluation
- Providing appropriate operational and support agency representation in the requirements and modification process
- Establishing and prioritizing modification requirements
- Fleet-wide interoperability and commonality
- Identification of weapon system funding requirements

AF PD 10-9 further states “if circumstances change or new information is available concerning a system, the user should bring the new data to the lead command which has the obligation to revisit the priority.”

By providing “appropriate operational and support agency representation”, the lead command brings knowledge of operations and field maintenance of current systems to the development effort and can project that knowledge to imply how future systems are likely to be operated and maintained. Their contact with operational units also puts them on the forefront of understanding evolving user needs.

The lead command, or user, role and unique knowledge is presented in Table 2.1.

User Characteristics Related to Design
• Role: define and prioritize requirements over the life cycle of the system; advocate funding
• Unique knowledge: operation of current systems; operational and maintenance considerations for future systems; evolving user needs

Table 2.1. User characteristics

2.2.1.2 System Program Office (Government Acquisition Agency)

The System Program Office (SPO) is led by a SPO director, whose responsibilities are laid out in Air Force Instruction 63-101 (AFI 63-101, 1994), which is entitled “Acquisition System.” As of this writing, a new draft of AFI 63-101 was in the

review cycle. Both the old version and the new draft of the document reaffirm the following basic responsibilities of the SPO director:

- Plan the acquisition strategy and management approach
- Execute the program within established cost, schedule and technical constraints
- Manage the program within established policies and procedures
- Establish and maintain a direct line of communication with using and acquisition commands and the operational test agency

Due to their responsibilities to ensure program constraints are met and all applicable laws and regulations are being followed during an acquisition, the SPO is required to possess detailed knowledge of program constraints and acquisition policies and regulations. Established policies and procedures include emphasis on such areas as supportability over the life cycle (sometimes referred to as "sustainment"), interoperability with other systems, and upgradeability (Interim Defense Acquisition Guidebook, 2002). These and similar other considerations are often thought of collectively as "ilities". They represent holistic, long-term factors other than system capabilities. The user community is typically more focused on system capabilities that meet their identified needs rather than on the full range of "ilities." The prime contractor follows the tasking identified by the SPO in the contract. The SPO is therefore uniquely positioned to act as a check and balance to ensure these aspects of the design are given proper consideration.

SPO characteristics are summarized in Table 2.2.

SPO Characteristics Related to Design
• Role: Plan, manage and execute the program within established policies and constraints • Unique knowledge: constraints, acquisition policies and regulations, and “ilities”

Table 2.2. SPO characteristics

2.2.1.3 Prime Contractor (Industry Lead)

The prime contractor is responsible for complying with the terms and conditions of the contract in the course of performing the development work for the new system. The contract includes technical requirements for the system and programmatic (cost and schedule) constraints. Per DoD guidance, “contracts shall include a strict minimum number of critical performance criteria (i.e., threshold and objective requirements) to allow industry maximum flexibility in meeting overall program objectives.” (Interim Defense Acquisition Guidebook, 2002) The contractor is therefore expected to transform top-level requirements into a detailed design.

Because of its role in generating a design to satisfy stated requirements, the prime contractor must have an understanding of the required technology and of possible design options. In the course of executing the development program, the contractor is the most knowledgeable party regarding cost, schedule and technical performance considerations of potential design choices. Contractors also contribute knowledge of other design factors including reliability, maintainability and future upgradeability of the system.

Contractor Characteristics Related to Design
• Role: generate design to meet overall program objectives within established constraints • Unique knowledge: technology; cost, schedule and technical performance implications of design options; "ilities"

Table 2.3. Contractor characteristics

2.2.2 The Design Phase

This research focuses on the design phase of Air Force acquisition. The primary guidance to Air Force program managers (PM) concerning the design phase comes from an Office of the Secretary of Defense (OSD) regulation formerly designated as DoD 5000.2-R and renamed "Interim Defense Acquisition Guidebook" as of 30 October 2002. The foreword for this document indicates it should be used "for best practices, lessons learned, and expectations, until replaced."

Chapter 5 of the guidebook is entitled "Program Design." The majority of the chapter involves a section called "systems engineering." This section starts with the following passage:

> The PM shall implement a sound systems engineering approach to translate approved operational needs and requirements into operationally suitable blocks of systems. The approach shall consist of a top-down, iterative process of requirements analysis, functional analysis and allocation, design synthesis and verification, and system analysis and control. Systems engineering shall permeate design, manufacturing, T&E, and support of the product. Systems engineering principles shall influence the balance between performance, risk, cost, and schedule.

The ensuing description indicates that the systems engineering process shall: "transform approved operational needs and requirements…into an integrated system design solution through concurrent consideration of all life-cycle needs (i.e., development, manufacturing, T&E, deployment, operations, support, training, and disposal). In these sections, T&E refers to test and evaluation.

The description of "requirements analysis" in the guidebook states, "the PM shall work with the user to establish and refine operational and design requirements. Together, they shall determine appropriate operational performance objectives, within affordability constraints."

And lastly, the following passage provides the intended purpose of design synthesis:

> Design synthesis translates functional and performance requirements into design solutions that include alternative people, product, and process concepts and solutions, and internal and external interfaces. Design solutions shall be sufficiently detailed to verify that open system performance requirements have been met.

Based on this DoD guidance, the design phase may be thought of as starting with a set of "operational needs and requirements" and ending when a system design is defined to address these requirements. The verification process that ensures the design meets requirements takes place primarily in a T&E phase after the design has been defined. Generating the design includes exploring trade spaces, selecting implementation approaches, managing interfaces, and balancing risk with cost, schedule and performance considerations. The contractor is therefore integrating and synthesizing a multitude of considerations to provide specified capabilities and system characteristics within programmatic constraints.

For most programs, the design phase culminates with a Critical Design Review (CDR) or equivalent meeting. While CDR and other formal reviews are no longer required to be held in compliance with military standards, one description of the intent of this program milestone appears in MIL-STD-1521B, "Technical Reviews and Audits for Systems, Equipments, and Computer Software" (DoD, 1986). This military standard indicates that a CDR shall be conducted "when detail design is essentially complete." The document cites the following purposes for the CDR:

- Determine that the detail design satisfies requirements of the development specifications.
- Establish compatibility of the design with other systems, equipment, facilities, software and personnel.
- Assess risk areas (cost, schedule and technical)
- Assess producibility of system hardware
- For software, determine the acceptability of the detailed design, performance, and test characteristics of the design solution, and adequacy of operation and support documentation.

At a final program CDR, both subsystem design and integration of the design at the system level should ideally be complete. For the case studies described in Chapter 5, the design phase was assumed to start with firm definition of system requirements and end at the last program CDR (or equivalent meeting) or shortly thereafter if there were residual design issues. The programs that were studied did not always have clean,

traditional starting and stopping points to the design phase, and any unique circumstances are described in the case study descriptions in Chapter 5.

Table 2.4 summarizes the top-level aspects of the design phase that are of relevance to this research.

Design Phase Characteristics
• Input: operational needs and requirements • Design starting milestone: Systems requirements review or equivalent • Design process: translate requirements into design solutions • Output: an integrated system design satisfying requirements and constraints • Design completion milestone: Critical design review or equivalent

Table 2.4. Design phase

2.2.3 Current Acquisition Policies

Recent policy memoranda from DoD and the Air Force provide insight into a growing emphasis on stakeholder collaboration, responsiveness to user needs, and faster development cycle times. The most significant of these memos are summarized below. Taken together, these statements from leadership and the ongoing revision of DoD and Air Force acquisition regulations make it clear that major shifts in acquisition-related policy are underway.

An Air Force headquarters policy memorandum entitled "Open Communication with Industry" (23 Jun 97) emphasized the importance of "open, fair and continuous communication" with industry "throughout all phases of the requirements and acquisition processes." The memo concludes, "The payoff will be streamlined source selections, reduced cycle time and well informed best value decisions."

The Air Force Chief of Staff and Secretary signed out a memo (11 Mar 02) entitled "Agile Acquisition and Logistics Transformation Imperatives" indicating that "unpredictable, asymmetric threats…demand fundamental changes in the way we conceive, acquire and sustain our capability." The memo goes on to say, "We must build strong, enduring partnerships between our warfighters, acquisition and sustainment professionals, so that our warfighters have the tools they need to fight and win wars." Some of the imperatives from the memo include the following:

- Change the way we work. Require the Headquarters Air Force Staff and MAJCOMs to reengineer their processes to reduce cycle times…
- Establish accountability. Performance standards must motivate agility, urgency, discipline and collaboration.
- Establish collaborative spiral requirements and development as the preferred approach.

The DoD lead acquisition executive issued a memo (12 Apr 02) with the subject heading "Evolutionary Acquisition and Spiral Development." The opening sentence indicates that the DoD has "established a preference for the use of evolutionary acquisition strategies relying on a spiral development process." The memo provided definitions of these two concepts and indicated that they are "methods that will allow us to reduce our cycle time and speed the delivery of advanced capability to our warfighters." They also "allow insertion of new technologies and capabilities over time" and are "focused on providing the warfighter with an initial capability which may be less than the full requirement as a trade-off for earlier delivery, agility, affordability, and risk reduction."

As the memo explains, evolutionary acquisition (EA) involves delivering a core capability and following up with future increments of capability over time. The DoD definition mentions that in some cases, “the ultimate functionality cannot be defined at the beginning of the program, and each increment of capability is defined by the maturation of the technologies matched with the evolving needs of the user.” Clearly, EA can be a means of adapting to changing circumstances more effectively than was possible in traditional, single delivery acquisition strategies.

The memo defines spiral development (SD) as “an iterative process for developing a defined set of capabilities within one increment.” It continues, “This process provides the opportunity for interaction between the user, tester and developer. In this process, the requirements are refined through experimentation and risk management, there is continuous feedback, and the user is provided the best possible capability within the increment.”

Table 2.5 provides the definitions of these two concepts. In summary, EA is an acquisition strategy for incremental delivery, and SD is a development strategy that facilitates iterative collection of customer feedback within a single deliverable increment.

Definitions of Evolutionary Acquisition and Spiral Development

- **Evolutionary Acquisition:** An acquisition strategy that defines, develops, produces or acquires, and fields **an initial hardware or software increment** (or block) of operational capability…capabilities can be provided in a shorter period of time, **followed by subsequent increments**…allowing for full and adaptable systems over time.
- **Spiral Development:** An iterative process for developing a defined set of capabilities **within one increment**…provides the opportunity for **interaction between the user, tester and developer**…requirements are refined through experimentation and risk management, there is continuous feedback, and the user is provided the best possible capability within the increment.

Source: Under Secretary of Defense (Acquisition, Technology, and Logistics) memo, 12 Apr, 2002, (emphasis added)

Table 2.5. Definitions of evolutionary acquisition and spiral development

Taken together, EA and SD represent means of enhancing collaboration, adaptability, and responsiveness to warfighter needs. However, it should be noted that this direction is relatively recent and stakeholders are still sorting out the best means for implementing the policies and accomplishing their intended objectives.

A June 04, 2002 memo entitled "Reality-based Acquisition System Policy for all Programs" (Sambur, 2002) was described in Chapter 1. Its emphasis was on reducing acquisition cycle time and meeting warfighter expectations. It emphasized the importance of collaboration and the preferred use of an EA strategy using an SD process.

Lastly, a Joint Staff memo entitled "Changes to the Requirements Generation System" was released on 7 October, 2002. It indicated, "The current process frequently produces stovepiped solutions that are not necessarily based on the future capabilities required by the joint warfighter." In order to promote a capabilities-based focus, the memo cancels the previous Mission Need Statement document in favor of a "mission area focused and capabilities-based document" to be described in a future Joint Staff instruction. It cites the need to coordinate with acquisition community changes in DoD regulations to "implement a more integrated and collaborative requirements and acquisition process." The memo reinforces the recurring theme of developing closer collaboration spanning requirements activities in the user community and acquisition activities.

Drafts in 2003 of new versions of DoD acquisition regulations, a new draft of the primary Air Force acquisition instruction (AFI 63-101), and the above-mentioned policy letters makes it clear that the current leadership, both in DoD and the Air Force, are serious about revamping the acquisition process to foster greater collaboration, reduce cycle times and improve responsiveness to evolving warfighter needs. The next challenge will be for personnel in the stakeholder organizations to come to grips with the necessary changes in their activities and processes to effectively implement these objectives. As this research effort sought to understand stakeholder collaboration and its relationship to adaptability, the subject of stakeholder roles became of central importance.

2.3 Insight from Defense Acquisition Reform Literature

The literature review conducted in the area of defense acquisition is summarized in the following paragraphs, with particular emphasis on the acquisition environment and interactions between stakeholders.

Peck and Scherer published a seminal work in 1962 at the completion of a Harvard University "Weapons Acquisition Research Project" that involved 27 corporations and numerous government agencies. The purpose of this research project was "to determine the nature of the relationships between the government and weapons contractors in the acquisition of advanced weapons and to analyze the effects of these relationships on weapons performance and the speed and cost of their acquisition." Major observations from Peck and Scherer's work include the following:

- Technical complexity of modern weapon systems leads to "internal uncertainties of the weapon acquisition process", which are the "unpredictability of time, cost and quality" of the systems being procured.
- "External uncertainties" arise from sudden and unpredictable changes in "technology, enemy plans, and our own defense policies." The authors note, "external uncertainties, perhaps more than internal uncertainties, may explain the variances of actual program outcomes from original predictions."
- "The changing requirements of weaponry require planning for change and flexibility. Planning for change is almost a contradiction in terms. The common notions on industry planning relate largely to obtaining efficiency in producing

well-established products, so that little experience is available to guide planning for change."

- Inter-functional conflict arises between research and development (R&D) organizations and operational users. R&D personnel tend to "emphasize sophistication and perfection of the product", which is in conflict with operating organizations who want "to have new equipment as soon as possible." However, operators also have "a propensity to request the inclusion of operational features which complicate both development and production."
- Establishing a "project group to integrate the diverse functional interests" is cited as a valuable practice. This concept is an early precursor to the modern concept of Integrated Product Teams, and reinforces the importance of stakeholder collaboration.

Peck and Scherer acknowledged the difficulty in reforming defense acquisition due to its unique nature, and they emphasized the need for flexibility and the exchange of reliable information in order to make effective acquisition decisions.

The "Packard Commission" report (Packard et al, 1986) was initiated by President Reagan to look at a wide range of defense issues. The foreword provides a key insight: "human effort must be channeled to good purpose through sound centralized policies, but free expression of people's energy, enthusiasm, and creativity must be encouraged in highly differentiated settings. Excellence can flourish only where individuals identify with a team, take personal pride in their work, concentrate their unique efforts, develop

specialized know how and above all constantly explore new and better ways to get their job done." Some of the commission's recommendations and observations are as follows:

- Set up centers of excellence - program manager and industry teams working closely together on new prototype weapons.
- Excellence does not come from legislation or directive. Rather, responsibility and authority should be put in the hands of working level people who have knowledge and enthusiasm for the tasks. Restore a sense of shared purpose and mutual confidence among Congress, DoD and industry.
- Congress should focus on larger issues such as overall defense posture and military performance - don't legislate minute details of DoD operations.
- DoD shouldn't measure quality by regulatory compliance. There are too many management layers, large staffs, and regulations. Reduce all of these. Adhere to basic, commonsense principles. Give a few capable people authority and responsibility to do the job, maintain short lines of communication, and hold people accountable for results.
- Defense contractors and DoD must each assume responsibility for improved self-governance. What is needed is an honest partnership, not legions of auditors.

The Packard Commission report is strongly in favor of empowerment, communication and close, trusting relationships in lieu of regulation and inspection.

Wilson (1989) writes about government bureaucracies and the reasons for their behavior. Regarding government employees, he notes, "it is hard to imagine why…personnel would run the risk of making "subtle" (and thus hard to defend)

judgments instead of following the rules in the most literal fashion. Also, "managers (and employees generally)…become averse to any action that risks violating a significant constraint." Wilson points out that Congress plays a significant role through legislation in ensuring fair treatment as well as responsive treatment to particular groups (small businesses, minorities, etc.) This legislation has the legitimate value of achieving Congressional objectives, but it does impose inefficiencies on government organizations tasked with ensuring all such constraints are satisfied. In the risk-averse environment created by a regulation and inspection culture, the challenge of encouraging innovation is even greater than is commonly encountered in large organizations in the private sector.

Fox (1988) indicates that every defense secretary since the establishment of the position has made a commitment to efficient management of acquisition, but with limited results. Defense programs are not developed and produced by a free market, so incentives are different and more problematic. Fox makes the following points:

- The government program manager relationship with the contractor should not be adversarial but a business relationship, a partnership perhaps. It tends to be characterized by rigorous bargaining, accompanied by tenacious regard for the best interests of one's own side.
- Management structure for defense acquisition must contain: a single chain of authority; a program manager with clear authority over horizontal layers; a single, well defined monitoring and auditing agency; and a single, well defined channel for Congressional inquiry and oversight.
- Government program managers are put in dual roles that are in conflict - program advocates and guardians of public funds. This situation provides incentive to

promote programs, postpone identification of problems, and seek additional funds over time.

- Incentives currently in place create an inverted system of rewards and penalties. Contractors are rewarded for higher costs with more profit. Higher priority is given to begin a new program or get more money than to control costs for existing programs. New contract forms, better planning, control and reporting systems, improved cost estimating, and change control systems are needed. However, it is unlikely that these changes would be effective unless government managers are skilled in implementation and the use of tools, and are rewarded for effective implementation.
- Why is there so much resistance to change? We are missing willingness to make lasting improvements in careers and reward/penalty systems. We must have external pressures – from Congress or the public - not just more funds. Without a sustained sense of urgency, imperatives for reform are unlikely to be successful.

Fox (1988) believes changes could be worth $40 billion a year. His primary insight is that understanding existing incentives and providing a sense of urgency are important considerations to implement change in the defense acquisition system.

Gregory (1989) emphasizes that “over management” by the Pentagon and Congress is a major problem, leading to lengthening of the development cycle. Excess paperwork and delays are costing billions, while the scandals that have led to legislation and emphasis on monitoring to prevent fraud involve miniscule amounts of money in comparison.

The Carnegie Commission on Science, Technology, and Government chaired by former Secretary of Defense William Perry, issued a report in August 1990, and a revised second edition in May 1993, entitled "New Thinking and American Defense Technology." The report notes, "Technology will be one of the nation's chief hedges against the uncertainties of the future." However, the difficulty associated with "selecting, procuring, and managing this technology" is increasing. The commission advocated replacing "milspec standards with dual military-industrial standards" and other measures designed to reduce bureaucracy imposed on defense contractors and increase purchase of commercial products. Spurred by this report, much of the substantive reform in DoD acquisition in the 1990's involved reducing the "red tape" and overhead associated with many of the formalized procedures and standards used in defense acquisition in the past.

From the standpoint of this research, the implications of empowering the prime contractor to determine many of the processes and procedures to be employed in developing a new system marks a turning point in the SPO-contractor relationship. This concept has blossomed over the past few years into a more results-based contracting approach in which the government is far less prescriptive, opening the door to greater innovation by industry.

Table 2.6 summarizes some of the main points from these reform writers in two major categories – the acquisition environment and relationships between stakeholders.

Author	Acquisition Environment	Stakeholder Relationships
Peck and Scherer (1962)	• Internal uncertainties due to technology complexity: time, cost and quality are unpredictable. • External uncertainties – changes in technology, enemy plans and defense policies • Changing requirements necessitate ability to plan for change (not a well-developed discipline)	• Tensions between R&D perfectionists and operators who want systems ASAP. • Operator tendency to want to add capabilities. • Multi-function teams help resolve inter-functional tensions. • Exchange reliable information to support effective decisions.
Packard Commission (1986)	• Too many management layers, large staffs, regulations and inspectors. • Congress is legislating "minute details"	• Encourage SPO – contractor teaming, honest partnership. • Focus authority, responsibility and accountability at lower levels • Maintain short lines of communication.
Wilson (1989)	• Rules-based environment makes government personnel risk averse and discourages innovation. • Congressional legislation adds constraints	
Fox (1988)	• Current incentives reward high costs (contractor) and starting new programs (government) rather than cost control. • No sense of urgency for reforms. • PM tension between being program advocate and guardian of public funds.	• Need SPO-contractor partnerships – business relationship vs. adversarial. • Streamline lines of communication.
Gregory (1989)	• Congressional and DoD oversight is causing excess paperwork and delays.	
Carnegie Commission (1990 and 1993)	• Technology is a hedge against future uncertainty. • Need less "red tape" and more commercial practices and purchases.	• Empower contractors to determine methods to achieve government-specified results.

Table 2.6. Summary of defense reform literature

The acquisition environment can be characterized as highly constrained and yet highly uncertain. Pressure on the Air Force and industry to ensure that laws and regulations are followed encourages a centralized management philosophy that is effective at enforcing constraints and facilitating coordination, but is less suited to encouraging innovation and responsiveness to uncertainty. (Wilson, 1989)

The literature highlights tensions and differences between stakeholders and advocates open communication and trusting relationships.

These insights regarding the acquisition environment and stakeholder interactions are revisited in Chapter 3 as part of the construction of a research design to study stakeholder collaboration in this unique, complex environment of Air Force acquisition.

2.4 Exploratory research

As a precursor to this study, the author interviewed 23 project managers in SPOs at the Air Force Electronic Systems Center (ESC) about their collaboration with users and contractors. The instrument for these interviews is provided in Appendix A. The interviews included questions on barriers and enablers to collaboration and on collaboration mechanisms. This data was used to help formulate the concept of system representations, which will be introduced below, and to help determine aspects of the research design including research questions, variables and research instruments.

Table 2.7 presents a summary of the primary barriers to collaboration that were cited most often during the interviews.

User Barriers to Collaboration	Contractor Barriers to Collaboration
• User lacks SPO perspective • Mixed priorities • Late buy-in • User understaffed • Lack of funds • SPO lacks user perspective	• Lack of commitment • Lack of open communication • Sole source led to risk aversion and lack of innovation • Competition limited information exchange • Corporate culture resisted certain technologies

Table 2.7. Barriers to collaboration

Table 2.8 covers the enablers that were identified. Given limitations in the data collection, it was not possible to differentiate between the significance of the different barriers and enablers. However, the data provides factors that the SPO felt were influences on collaboration.

User Enablers of Collaboration	Contractor Enablers of Collaboration
• Leadership emphasis on collaboration • Shared sense of mission • Healthy level of funding • Constant communication with many mechanisms • Consistent, empowered user representatives • Use of common tools/language • Liaison/exchange of personnel • Previous collaboration • Time-related forcing functions • Interpersonal factors (trust, personality, cooperation)	• Leadership emphasis on collaboration • Future business or technology needs • Communication mechanisms • Experience, continuity and expertise • Contract mechanisms • SPO management philosophy • Co-location • Shared goals • Interaction of technical personnel • Interpersonal factors (honesty, cooperation, recognition, pride)

Table 2.8. Enablers of collaboration

Another set of data was collected from the SPO managers indicating what collaboration mechanisms were used to interact with the user and the contractor, and in particular, which mechanisms were of particular importance in creating shared understanding of the system under development. In all, 28 separate collaborative mechanisms were identified. Table 2.9 summarizes the results. This list is in order with the most frequently mentioned mechanisms coming first. The number in parentheses indicates how many of the 23 program managers cited the mechanism.

<table>
<tr><td>Collaboration Mechanisms (Important)

• Informal conversations (15)
• Tech Interchange Meetings/Informal meetings (12)
• Electronic documents (8)
• Software demonstrations (8)
• Briefings (8)
• Prototypes (6)
• Integrated Product Teams (IPTs)/Working Groups (6)
• Trade studies (5)
• Telephone conferences (5)
• Cost models (5)
• Test (4)</td><td>Collaboration Mechanisms (Used)

• Informal conversations (23)
• Technical Interchange Meetings/Informal meetings (21)
• Electronic documents (20)
• Web (17)
• Test (16)
• Story about past projects (15)
• Cost models (14)
• Trades (12)
• Drawings (11)
• Briefings (8)
• Software demonstrations (8)
• Analysis tools (8)
• Prototypes (7)
• Trade studies (7)
• IPT/Working Groups (6)
• Telephone conferences (6)</td></tr>
</table>

Table 2.9. Collaboration mechanisms

Of the collaboration mechanisms used, Table 2.10 lists those that were listed as important to creating shared understanding in the highest percentage of cases. For

example, informal communications were critical to 15 of the 23 programs that used them, or 65%, while prototypes were critical to 6 out of 7 of the programs, or 86%.

Collaborative Mechanism	**# Programs Using**	**# Programs Critical**	**% Important for Shared Understanding**
• Briefings	8	8	100%
• Software demonstrations	8	8	100
• IPTs/Working Groups	6	6	100
• Prototypes	6	7	86
• Telephone conferences	5	6	83
• Informal conversations	15	23	65
• Technical Interchange Meetings /Informal Meetings	12	21	57
• Trade studies	5	12	42
• Electronic documents	8	20	40
• Cost models	5	14	36
• Test	4	16	25

Table 2.10. Important use percentage for collaborative mechanisms

Table 2.10 showed that a variety of mechanisms were considered valuable for gaining a shared understanding of the program. Two entries near the top of the table were software demonstrations and prototypes, both of which provided a shared representation of the system design for stakeholder evaluation and feedback.

Comments in the interviews reinforced the insight that some mechanisms provided a representation of the system, and that these mechanisms were powerful tools in creating a shared understanding between stakeholders. This theme of a "system representation" as a collaboration mechanism is central to this research and will be developed further in Chapter 3.

2.5 Summary

The review of stakeholders highlights the differences between the roles and knowledge of the primary participants during planning and execution of the design phase. The SPO-user interface is formalized due to the necessity of requirements generation and refinement, and the SPO-contractor relationship is formalized by a contractual relationship. However, the relationship between the user and the contractor is not formally established and is therefore potentially more variable between programs.

Figure 2.1 consolidates information on the roles and knowledge of the stakeholders, aspects of their interaction, and key observations about the acquisition environment, as established in this chapter. The boundaries of design, starting with requirements definition and ending after resolution of design issues, are also illustrated.

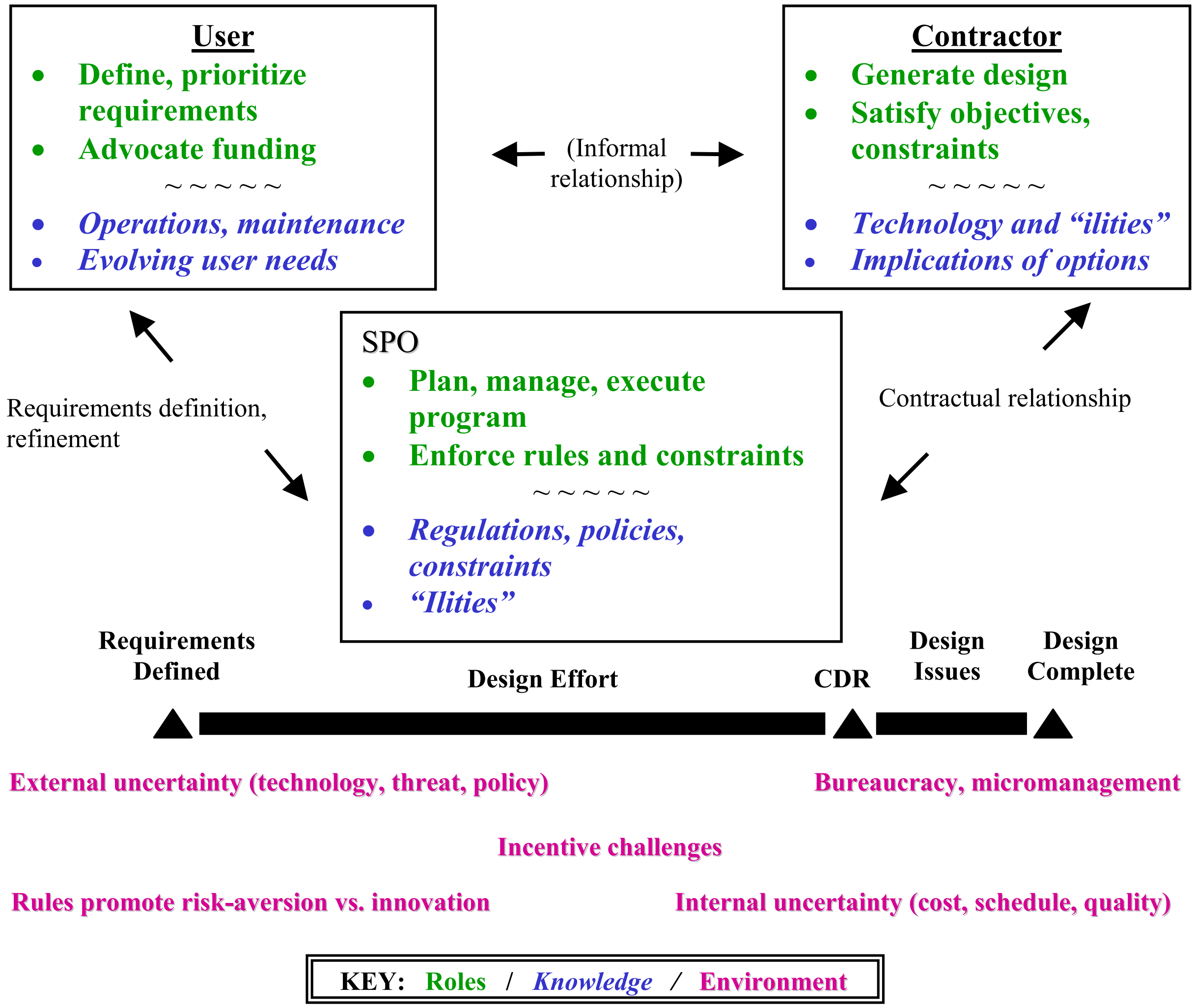


Figure 2.1. Stakeholders and the design phase

The collective information in this chapter provides a specific, real-world context for this research. This context facilitates a focused study of the phenomena of stakeholder collaboration and adaptability during the design phase of Air Force weapon system acquisition. Chapter 3 will develop theoretical lenses that will contribute further to formulation of specific research questions.

Chapter 3 Application of Theory to Program Adaptability

3.1 Introduction

Collaboration between stakeholders can be thought of as interaction that brings about results that would not be observed if the stakeholders functioned in isolation. This perspective can be explored by applying a body of thought that is broadly referred to as "systems theory." This chapter starts with a brief exploration of systems theory concepts to develop a way of thinking about stakeholder collaboration in which "the whole is greater than the sum of the parts."

The chapter then explores a derivative of systems theory that has gained prominence in the last two decades -- complex adaptive systems (CAS) theory. CAS theory emphasizes interaction of agents and the phenomenon of emergent order. Although several writers have applied CAS theory to the organizational context, this application of CAS theory is still relatively nascent. Taken as a whole, CAS theory provides insight into why and how interaction of the stakeholders described in Chapter 2 may be of significance in promoting program adaptability during new system design.

The operative resource during the design phase is information, and yet the flow of information across organizational boundaries, as for example between the user, SPO and contractor, can be problematic (Carlile, 1997.) The concept of boundary objects, and particularly a special category of boundary objects referred to as system representations, offers an approach to bridge knowledge boundaries between stakeholders. The significance of system representations and stakeholder collaboration as means to address some of the considerations raised by CAS theory are discussed.

The theoretical considerations developed in this chapter are combined with elements of the Air Force context described in chapter 2 to formulate a set of research questions. These questions were used to guide the creation of the research method that will be presented in Chapter 4.

3.2 Systems Theory

The concept of a system involves both the parts of the system and the interaction of the parts. Rechtin (1991) defines a system as "a collection of things working together to produce something greater." He notes the unique elements of a system are the relationships between its parts. Rechtin explains that systems can be thought of as unbounded in that each system is inherently part of a still larger system. However, defining a boundary provides a way of focusing on a system at a desired level of functionality. Such a boundary allows differentiation between the system and external elements that can be thought of as the system's environment.

Steward (1995) defines a system as, "A collection of parts and relations between the parts such that the behavior of the whole is a function not only of the behaviors of the parts, but also of the relations among them." This definition again emphasizes that interaction of the parts has significance for the system and its behavior.

For a purposeful system such as an organization or set of organizations, the system may be thought of as having a particular function, such as raising money for charity or providing national defense. The system's function or functions arise both from the functions of the component parts and from the interactions of the components.

Summing up these observations, a system can be thought of as having the following properties:

- It performs a function
- It is composed of components (that may or may not individually provide functionality)
- Interaction of components provides functionality above and beyond the functionality of the separate components
- It has a boundary
- It has interactions with an environment external to its boundary

Applying this thought process to the Air Force design phase, it is possible to think of the three primary stakeholders described in Chapter 2, along with their interactions, as making up a system that is generating the weapon system design. This research defines the system of interest as having the properties listed in Table 3.1.

- Function: to design a new weapon system meeting the needs of the warfighter within programmatic constraints.
- Organizational components: the warfighter, the System Program Office (SPO), and the contractor
- Interactions between components (top level):
  - The warfighter defines and refines requirements with the SPO
  - The SPO has a contractual relationship with the contractor to meet program objectives within constraints
  - The contractor generates the design for government evaluation
  - Stakeholders share knowledge, leading to identification and disposition of issues and opportunities
- Boundary: includes components; does not include organizational environment.
- Organizational environment: interfaces with government oversight (Air Staff, DoD, Congress, the Administration), subcontractors, contractor shareholders, functional offices (test and evaluation, budget, logistics), taxpayers, and other military services.

Table 3.1. Design system properties

In this context, inter-organizational interactions take the form of stakeholder collaboration. The design process, as discussed in Chapter 2, represents a transformation of requirements into a detailed design that satisfies objectives and constraints. Requirements take the form of information about what the system must do and what

characteristics it must have. The design consists of information describing how technology (hardware and/or software) will be integrated into a functional system. Therefore, the nature of the task of designing a system puts the primary focus of collaboration on information and information sharing.

In the next section, this foundational framework will be extended using complex adaptive systems theory to include concepts that influence adaptability. In this research, adaptability refers to a program's capacity to change requirements or design as a result of stakeholder collaboration.

3.3 Complex Adaptive Systems Theory

In the Newtonian world of deterministic thinking, the study and design of systems has focused on predictability. Disciplines like systems engineering make use of deterministic approaches such as decomposition of systems in an effort to achieve understanding and control. However, many systems adapt over time, exhibiting emergent characteristics that are difficult or impossible to predict. Complex adaptive systems (CAS) theory is a rapidly developing school of thought that seeks deeper understanding of such systems. This section describes CAS theory and generates a set of CAS principles that pertain to this research.

Belgian Nobel laureate Ilya Prigogine was one of the seminal thinkers in this area, before CAS became a recognized concept. He investigated the sources of order and structure in the world. Prigogine (1984) observed that atoms and molecules are exposed to energy and material flowing in from the outside, partially reversing the decay required

by the second law of thermodynamics. As a result, systems are able to spontaneously organize themselves into a series of complex structures. This work represented some of the early thinking on self-organization of systems, a key CAS precept.

The rise of "complex adaptive systems" as a school of thought started in the mid-1980's with the formation of the Santa Fe Institute, a New Mexico think tank formed in part by former members of the nearby Los Alamos National Laboratory. Their members included several Nobel laureates and many other leading thinkers in a diverse set of fields that included economics, physics, biology, ecology and archaeology.

In his book, "Complexity: the Emerging Science at the Edge of Order and Chaos", M. Mitchell Waldrop (1992) describes the rise and expansion of the Santa Fe Institute and summarizes the insights of its most prominent members on the subject of complex adaptive systems. Waldrop explains the objectives associated with the development and use of CAS concepts. Santa Fe members sought to pursue a common theoretical framework for complexity and a means of understanding the spontaneous, self-organizing dynamics of the world.

CAS have several common characteristics that recur in a number of natural and human contexts. The most commonly noted characteristics in the CAS literature are as follows (Waldrop, 1992; Kauffman, 1995; Holland, 1995):

- CAS are composed of a network of many agents gathering information, learning and acting in parallel in an environment that is influenced by the interactions of these agents.
- In an aggregate environment referred to as a fitness landscape, agents are adapting as they strive to find the highest peak, or fitness level.

- CAS are balanced between order and anarchy, at the edge of chaos. As Waldrop (1992) describes, "…frozen systems can always do better by loosening up a bit, and turbulent systems can always do better by getting themselves a little more organized. So if a system isn't on the edge of chaos already, you'd expect learning and evolution to push it in that direction…to make the edge of chaos stable, the natural place for complex, adaptive systems to be."
- CAS are self-organizing. Order is emergent instead of pre-determined, always unfolding and always in transition, creating a condition of perpetual novelty.
- CAS tend to exist in many levels of organization in the sense that agents at one level are the building blocks for agents at the next level. An example is cells, which make up organisms, which in turn make up an ecosystem.
- Finally, CAS, by their nature, have a future that is structured but hard to predict.

Examples of CAS are widespread. In the natural world, brains, immune systems, ecologies, cells, developing embryos, and ant colonies all fall under the category of CAS. In the human world, political parties, scientific communities and the economy are examples.

For purposes of this research, it was necessary to consolidate the CAS concepts listed above into key principles. The principles, listed in Table 3.2, capture the unique nature and behavior of CAS and represent insightful ways of thinking about how such systems adapt.

Key CAS Principles 1. Interaction: CAS are composed of a network of agents whose interactions give rise to self-organization 2. Self-organization: CAS systems display order that is emergent 3. Zone of novelty: CAS settle into a "zone of novelty" between order and chaos

Table 3.2. Key CAS principles

Numerous writers on complexity theory (Waldrop, 1992; Kauffman, 1995; Holland, 1995; Arthur, et al, eds., 1997; Axelrod 1997; Gell-Mann 1994, etc.) have expounded on these fundamental concepts. The following representative examples are provided from the fields of economics, biology and management to illustrate the robustness and significance of CAS principles across different disciplines. These same principles apply across a wide range of contexts that are documented in the CAS literature.

Principle #1: Interaction (economics)

Brian Arthur (Arthur, et al, ed. 1997), a leading theorist in the field of economics, co-edited the discussions and findings of an economic conference held under the auspices of the Sante Fe Institute in 1996. The conference was held for the purpose of developing a "complexity perspective" on economics and economic modeling. Arthur documents that the assembled group of economists recognized a difference between individual economic agents and the aggregate economic system that emerged from the agents' interactions. The interactions of these heterogeneous agents led them to self-organize into network-based structures that were not predisposed to settle into equilibrium.

The profound conclusion of the conference was that it would be necessary to move away from the equilibrium-based view of economics that had dominated the field

for decades. Due to the implications of interaction of agents in a complex adaptive system, the concept of equilibrium needed to be replaced with a view of the economy as a system that included the phenomenon of self-organization.

Principle #2: Self-organization (biology)

Stuart Kauffman, one of the most influential thinkers associated with the Santa Fe Institute, wrote "At Home in the Universe: the Search for the Laws of Self-Organization and Complexity" (1995) to explain his perspectives on complexity in the field of biology. Kauffman argues that a combination of natural selection and self-organization leads to matter organizing itself into complex structures in spite of the forces of entropy. Kauffman discusses the second law of thermodynamics, which has as a consequence the disappearance of order from equilibrium systems. This law leads to "our current sense that an incoherent collapse of order is the natural state of things." Yet Kauffman goes on to cite the abundant evidence of order in our world, from microscopic cells, to the plenitude of species unleashed in the Cambrian era, to "our postmodern technological era, in which the exploding rate of innovation brings the time horizon of future shock ever closer." These examples illustrate a fundamental principle of complex adaptive systems, their capacity for self-organization, or what Kauffman calls "order for free."

Principle #3: Zone of novelty (management)

The field of management also contains references to the significance of CAS theory. Richard Pascale (1999) wrote about CAS principles as demonstrated by the

efforts at Royal Dutch/Shell in the 1990's to apply complex adaptive system theory to business strategy. Pascale writes, "the lure of equilibrium is a constant danger to successful firms." He relates how Shell recognized that competitive threats necessitated "involving the front lines in renewal."

Shell managers found that "while leaders provide the vision and establish the context, solutions to ongoing challenges are generated by the people closest to the action." As Pascale describes, "novelty emerges in the space between rigidity and randomness." Shell coached teams on a "more direct, informal, and less hierarchical way of thinking" to create this space for novelty. These teams identified new market opportunities that helped Shell adapt to stay ahead of their competition.

In the context of Air Force development programs, as described in chapter 2, the agents in the system can be thought of as the three primary stakeholders – the System Program Office (SPO), the user and the prime contractor. As these agents interact, they are forging the future of the new system, which can sometimes take the system in new, unexpected directions. Examples could include incorporating knowledge from operation of existing systems, responding to a new or changed threat, or exploiting a new technology. Self-organization corresponds to adaptations that the stakeholders may decide to make in the requirements or design of the system. The "zone of novelty" in this context is created by stakeholder actions and interactions that promote adaptability.

These implications from CAS theory will be re-evaluated in an organizational perspective in the next section, which covers the most significant writers on the subject of organizations as complex adaptive systems.

3.4 CAS Theory and Organizations

Recently, several writers have applied CAS theory to organizations. The contributions of three such authors are summarized below. Up to the present time, published work on CAS and organizations has been based primarily on philosophical observations and individual experience in industry rather than scientific studies. Therefore, this body of work is of limited utility from the standpoint of predictive power. However, the organizational principles cited by the writers relate to the CAS principles described above, and provide some context for considering these CAS principles from the perspective of organizations. The result of this section is a set of CAS constructs that echo the basic CAS principles listed in Table 3.2, but have relevance to the context of organizations such as those of the stakeholders involved in the design phase of Air Force acquisition.

In her book, Leadership and the New Science: Discovering Order in a Chaotic World, Wheatley (1999) writes that we are hypnotized by the structures we create to help us hold back "the dark forces that threaten to destroy us", but in reality the world is inherently orderly. Fluctuations and change are essential to the process by which order is created. Wheatley indicates that the things we fear in an organization, such as disruptions, confusion and chaos, are necessary to awaken creativity.

The key concepts offered by Wheatley are summarized in Table 3.3.

- Self-reference - the need to focus on the identity of the organization, including its intent and vision
- Quality of relationships – by increasing participation, the organization benefits from more interpretations of information, developing a wiser sense of what is going on and what needs to be done
- Use of information - essential to notice newness and open up to potentials

Table 3.3. Summary of Wheatley concepts

The last concept in Table 3.3 refers to the skill of using information, not just to regulate, but also to notice anything new that might perturb stability. Organizations need to know how to stay acutely aware of what's happening now, and they need to create greater access to information, trusting that people know their jobs and the team's purpose, and will know what to do with it. If, on the other hand, the organization closes itself off from disturbances and change, it will atrophy and die. This concept of looking for and resolving potential perturbations to organizational stability is analogous to the CAS principle of "self-organization."

Stacey (1996) introduced the concept of control parameters in his book, Complexity and Creativity in Organizations. Stacey believes five key parameters enable organizations to balance between order and chaos in a zone of perpetual novelty where adaptive responses to change are possible. Stacey's control parameters are summarized in Table 3.4.

- Rate of information flow – it is necessary to flow information beyond formal channels to promote creativity
- Degree of diversity - diversity in ways of thinking promotes learning, although too much diversity can lead an organization to anarchy
- Richness of connectivity - few, strong ties between agents produces stable behavior (too little variety for effective learning), while many, weak ties produce unstable behavior (too much variety)
- Level of contained anxiety - only when the anxiety of work and creativity can be experienced, but also held and contained, is creativity possible
- Power differential - at a critical point, anxiety is contained through the comfort of clear structures in the organization, but sufficient freedom exists to express opinions and take risks without fear

Table 3.4. Stacey control parameters

Each of these parameters involves two extremes that represent, respectively, ordered and chaotic states. In between the extremes is a zone of novelty where structure and flexibility co-exist. The fundamental concept of looking for a zone between two extremes in which creativity flourishes is in concert with the CAS principle of a "zone of novelty" and provides a provocative way of looking at organizational factors that influence adaptability.

Highsmith (2000) in Adaptive Software Development: a Collaborative Approach to Managing Complex Systems draws on his decades of experience in software development to describe applications of CAS theory to the real-world context of software development. Many of his ideas have relevance to any complex organizational enterprise. Highsmith (2000) offers practical recommendations for managing complex development programs through collaboration. Table 3.5 summarizes his concepts.

- Active participation - go beyond shared information to a practice of shared creativity
- Create collaborative information structure – enhance the flow of information
- Information content and context –managing of information must address both
- Teams manage their structure and support – create a bottom-up network
- Time boxing to force periodic convergence –combine room for creativity with incentive (time deadlines) to achieve progress
- Release of partial work for learning – review by participants and customers elicits learning from successes and mistakes - the job can be "done right the last time"
- Balance structure and direction with freedom to express opinions and create

Table 3.5. Summary of Highsmith concepts

Highsmith focuses on management of the content and flow of information, including the use of partial information to promote learning. These concepts reflect the CAS principle of “interaction” between agents as applied to an organizational context in which the flow of information is the key form of interaction. Highsmith urges an emphasis on interaction through development of tools and processes for sharing information.

These three writers have a central core of common ideas related to adaptability, although they each adopt a different emphasis. Wheatley advocates embracing change as the source of emergent order. Stacey describes the need to position the organization in a zone of novelty to promote adaptability. Highsmith supplies practical considerations for effective collaboration through information sharing. Consolidating the concepts of these writers leads to the list presented in Table 3.6 of CAS organizational constructs that promote organizational adaptability.

Writer	Emphasis	CAS Organizational Construct	Underlying CAS Principle
Wheatley	Embrace change as the source of emergent order	Look for and resolve potential perturbations to stability	Self-organization
Stacey	Position organization in a zone of novelty	Balance between structure and flexibility	Zone of novelty
Highsmith	Effective collaboration through information sharing	Develop tools and processes for information sharing	Interaction

Table 3.6. CAS constructs for organizational adaptability

3.5 CAS Organizational Constructs and the Air Force Acquisition Context

This section assesses the constructs listed in Table 3.6 in the context of the primary stakeholders and the design process. The intent is to bridge from the theoretical constructs and CAS principles to practical considerations whose study may lead to insights for making Air Force programs more adaptable. The three constructs, as applied to the Air Force acquisition context, provide the foundation for the research questions defined at the end of this chapter.

3.5.1 Develop Tools and Processes for Information Sharing

Interaction between the user, the SPO and the prime contractor to share information across organizational boundaries presents difficulties. Carlile (1997) describes specialization of "knowledge in practice", i.e. differences in knowledge stemming from the practices associated with disparate functional groups. This aspect of knowledge leads to a need for knowledge to be transformed to "effectively deal with differences, dependencies, and the novelty present at a given (knowledge) boundary." (Carlile, 2002) Boundary objects, as will be discussed in section 3.6, provide a mechanism for transforming knowledge to address this concern. A "system representation" is a specific type of boundary object defined in this research in order to accommodate the circumstances of Air Force acquisition. Usage of a system representation (SR) provides a means for stakeholder interaction for the purpose of sharing information. The concept of a SR is developed in section 3.6.2.

3.5.2 Look for and Resolve Potential Perturbations to Stability

In the Air Force acquisition context, perturbations are factors that have the potential to change the cost, schedule or technical baseline of a program. They could include a variety of developments such as budget cuts, threat changes, availability of new technology, or parts obsolescence. Perturbations could also involve conflicting aspects of a development program such as requirements that are driving the cost above budget or government furnished equipment that will not be available to meet program schedules.

When knowledge pertaining to the identification and resolution of these perturbations is distributed among the stakeholders, knowledge sharing becomes important to restore program stability. To the extent that the requisite knowledge sharing includes aspects of the system design or potential changes to that design, a SR can play a part in manifesting this CAS construct.

3.5.3 Balance Between Structure and Flexibility

The eight case studies provide diverse examples of design efforts in which different stakeholder activities provided elements of structure and flexibility with varying results in terms of program adaptability. The study of stakeholder roles provides insight into how these organizations can create a balance in which innovative thought is encouraged while control of the level of program risk is still maintained.

3.6 Boundary Objects

Section 3.5.1 introduced the difficulty associated with sharing knowledge across organizational boundaries. Carlile (1997, 2002) has written about the practical application of "boundary objects" as a way to transform knowledge at a knowledge boundary. Boundary objects provide "a means of resolving the consequences that arise when different kinds of knowledge are dependent on each other." (Carlile, 2002) As an example, the acceptability of a contractor's software design to accomplish a task might be dependent on the Air Force user's intended operational concept for employing the system. Providing a boundary object that represented the design to the user could surface

differences of interpretation as to what constituted an operationally acceptable design implementation.

This section discusses boundary objects from a theoretical basis and then defines a specific example of a boundary object, referred to as a system representation, that has particular applicability to the Air Force acquisition context.

3.6.1 Definition and Critical Features

Boundary objects are representations of knowledge that can play a role in resolving the difficulties faced at knowledge boundaries between groups or organizations engaged in product development. Imagine a multi-functional design team that is trying to evaluate a product design in its early stages. As a simplistic example, suppose that the team has a design engineer, a marketing representative and a manufacturing engineer. These three functional entities have different expectations and concerns for the design. In this situation, a useful boundary object might be a physical or computer simulated representation of the design that captures the design engineer's partial design, but that also gives an indication of the manufacturing considerations and the appeal to the customer of the eventual, finished design. Such a boundary object, if it could be modified over time, would allow the multi-functional design team to identify issues and resolve differences in an iterative, interactive fashion before freezing the design. Going beyond this illustrative example, theorists (Star, Carlile) have constructed a broad definition of boundary objects.

Star (1989) defined boundary objects as, "objects that are both plastic enough to adapt to local needs and constraints of the several parties employing them, yet robust

enough to maintain a common identity across sites." Star notes that the boundary object works to overcome the problem of representing information to parties versed in different disciplines. Carlile (1997) offers the following explanation of how boundary objects work:

> By representing both the local needs of a given function while at the same time expressing common, cross-functional concerns, a boundary object works to expand the character of knowledge and the problem spaces of the various functions involved so as to create a larger, cross-functional problem space.

Carlile's "cross- functional problem space" refers to a working environment enabled by a boundary object in which information is available for different functional representatives to share in the course of identifying and resolving problems.

Star (1989) and Carlile (2002) refer to four categories of boundary objects. The categories, with representative examples, are listed in Table 3.7. "Models or objects" are of particular interest to this research since they "produce a representation or object that can be observed and then used by each functional setting." Carlile (2002) writes that boundary objects in this category "depict or demonstrate the current or possible forms, relationships, and functions of a product across functional concerns."

Categories of Boundary Objects	Examples
Repositories	Cost databases, parts libraries
Standardized forms and methods	Engineering change forms
Objects or models	Drawings, prototypes, computer simulations
Maps of boundaries	Gantt charts, process maps

Table 3.7. Categories of boundary objects

Carlile (2002) identifies three characteristics that make a boundary object more "useful in joint problem solving at a given boundary." These features are:

- It "establishes a shared syntax or language for individuals to represent their knowledge" at the boundary.
- It "provides a concrete means for individuals to specify and learn about their differences and dependencies across a given boundary."
- It "facilitates a process where individuals can jointly transform their knowledge." It must be possible to change the "object or representation used."

These three characteristics have a cumulative effect on the transfer, translation and transformation of knowledge. Without a boundary object, Carlile (2002) identifies that syntactic differences can lead to lack of understanding when knowledge is simply transferred across a boundary, as happens in an email or document. Therefore, a boundary object such as a hardware prototype can establish a shared syntax by translating knowledge into a form that is recognizable across boundaries. However, a boundary object also facilitates identification of differences and dependencies (Carlile, 2002), which, while highly desirable, leads to another concern. Carlile (2002) explains that knowledge must be transformed in order to help resolve these differences and dependencies. Boundary objects therefore must also be capable of being modified iteratively to search for solutions that satisfy all functional concerns. An acceptable solution modifies, or transforms, knowledge to accommodate the considerations of all parties.

In summary, boundary objects represent knowledge to participants from different disciplines, facilitating a sharing of knowledge, identification of differences, and resolution of these differences. These characteristics make boundary objects, and particularly the category of "objects or models", important in the context of Air Force acquisition, given the need to share information between disparate stakeholders. Users, SPOs and contractors have different needs, backgrounds and perspectives. The specific challenges associated with knowledge transfer between these stakeholders are discussed in the next section.

3.6.2 System Representations

System representations are a type of boundary object defined in this research to fit the circumstances of knowledge sharing faced by Air Force acquisition stakeholders during the design phase of new system acquisitions. This section describes the unique context that shapes the definition of a system representation.

The objective of the design process is to meet user requirements with a contractor-derived design solution, and to do so without violating program constraints. In this situation, a boundary object is effective to the extent that it can represent the contractor's design in a visual way so that user and SPO personnel can interact with the SR to evaluate whether the design meets their expectations. Stakeholder feedback amounts to an interim verification that the design is likely to meet requirements, as well as an interim validation that the right requirements have been levied and correctly

interpreted. Carlile's (2002) boundary object category of "objects or models" allows such a representation of the design.

The system representation consists of partial information. It provides a visual depiction of the contractor's design as it is envisioned at a certain point in time. Knowledge of user needs and SPO interest areas becomes more explicit when these two stakeholders provide feedback to the contractor's evolving design in an iterative fashion. If the SR can be modified to reflect stakeholder feedback, its effectiveness at supporting an iterative evaluation of the design will be enhanced. Also, if the SR is available early in the design phase, potential changes are likely to be easier to implement.

Highsmith's (2000) practical considerations for managing the content and flow of information, as laid out in Table 3.5, have relevance for the usage of system representations. Using a SR satisfies Highsmith's injunction to release partial work for learning. Highsmith recommends the practice of "time boxing" in which teams are allowed to investigate options creatively within the confines of schedule deadlines. These deadlines force tradeoffs and ensure progress. Time considerations are important for SR usage to ensure creative exploration of opportunities for change does not expand to the point of impacting program execution. The relevance of time considerations to SR usage will be revisited during the analysis of stakeholder roles in chapter 7.

Table 3.8 provides a definition, examples, characteristics for effectiveness, usage considerations and a primary purpose of system representations.

• Definition: A system representation (SR) is a visible, interactive representation of the contractor's system design as it is envisioned at a point in time. • Examples: prototypes, software beta releases, mockups, simulations • Characteristics for Effectiveness: it should be possible to modify the SR to allow for iterative stakeholder feedback; and the SR should be available early in the design phase to mitigate the impact of potential changes. • Usage considerations: provide opportunity for learning and creativity; use "time box" to force tradeoffs and ensure progress. • Primary purpose: The SR fosters shared understanding between stakeholders regarding the evolving design.

Table 3.8. System representation definition and characteristics

Applying boundary object concepts to acquisition of Air Force systems shifts the focus from Carlile's (1997, 2002) inter-functional boundaries to inter-organizational boundaries. However, this research will assume that the same concerns regarding the difficulty of knowledge transfer, translation and transformation described above apply to the Air Force acquisition context because of knowledge differences between the user community (operators), the SPOs (acquirers) and prime contractors (technology developers and integrators).

3.7 Research Questions

The fundamental motivation for this research is to discover how Air Force programs can improve their ability to adapt during the design phase of acquisition in order to provide more value to the warfighter. The application of complex adaptive systems theory to organizations has provided guiding constructs that may influence the capacity of organizations to be adaptable. A particular type of boundary object called a system representation (SR) was described as a way to address concerns with sharing information across stakeholder organizational boundaries. Taken together, these concepts provide material for construction of focused research questions.

Two aspects of SRs are important to program adaptability. Using a SR may provide a means of implementing the CAS organizational construct of looking for and resolving potential perturbations to stability. If a SR facilitates stakeholder identification and resolution of potential perturbations, it is effectively helping stakeholders to adapt. With this consideration in mind, the first research question asks how a SR enhances adaptability.

The next research question involves what makes an effective SR. This question relates to the CAS construct to develop tools and processes for information sharing. To the extent that SRs can be made effective at facilitating knowledge sharing, they can address this construct.

The roles of the stakeholders during design have an impact on the ability of a program to stay in an adaptive zone in which creative responses to issues and opportunities can be explored without creating unacceptable levels of program risk. This

consideration engages the third CAS construct involving balance between structure and flexibility. The actions and interactions of the stakeholders therefore have the potential to exert a strong influence on program adaptability. Stakeholder roles are the subject of the third research question.

Lastly, program characteristics represent alternate explanations of adaptability levels. Program characteristics include requirements uncertainty, funding level and duration of the design phase. The potential relationship between program characteristics and adaptability is explored in the fourth research question.

The research questions are listed in Table 3.9.

• How does a system representation enhance adaptability?
• What characteristics make system representations effective at promoting adaptability?
• What are the roles of stakeholders in facilitating program adaptability?
• Do certain characteristics of programs (requirements uncertainty, funding level and duration of design phase) predispose them to be more or less adaptable?

Table 3.9. Research questions

Chapter 4 will describe the research method developed and executed to address these questions.

Chapter 4: Research Method

4.1 Introduction

The primary focus of this research was on collaboration of three primary stakeholders - the System Program Office (SPO), the warfighter or user, and the contractor - during the design phase of Air Force development programs. After exploratory research and a review of Air Force regulations, the design phase was selected for study because stakeholder roles, and particularly the collaborative roles of the warfighter, were less clearly defined for this phase than for other phases such as requirements definition and system test. Also, program changes implemented during design were frequently based on consideration of both the "what's", or user requirements, and the "how's", or contractor design choices. This confluence argued for the importance of stakeholder collaboration to promote knowledge sharing as an enabler of adaptive decisions. Such decisions, implemented during design, were more affordable than in later stages of development.

This chapter describes the research method that was employed to collect data and provides rationale for the approach that was selected. Then the sampling strategy and the data collection process are addressed. Later sections of the chapter describe how data were collected and what variables were defined. The significance of stakeholder roles is discussed. The final topic for the chapter is validity of the method.

Three complementary approaches were used to collect data– questionnaires, program documentation reviews, and interviews. Data were analyzed to look for patterns and practices related to collaboration, knowledge sharing and adaptability. Also,

stakeholder roles and program characteristics were assessed regarding their potential influence on adaptability.

4.2 Method

The theme of stakeholder collaboration required a research method that would allow insight into the perspectives of three key stakeholders regarding Air Force development programs during the design phase. A broad based survey was not feasible. Both the diversity between programs and differing stakeholder perspectives would have made it problematic to define all-encompassing questions. Also, the variables involved in stakeholder collaboration and adaptability were not sufficiently understood to codify in a questionnaire. Both longitudinal studies and experiments were impractical to arrange, and either approach would have constrained the number of programs that could be observed to such a small number that the risk of non-representative results would have been high.

Historical case studies were selected as the best research method. This approach allowed the study of eight programs, providing diversity in the data set. The critical factor was availability of stakeholders with "corporate memory" of the design phase of their programs. In almost all cases, the primary individuals from each organization were still involved in follow-on phases of the program and were available for interviews. In the remaining cases, knowledgeable individuals were found who had cognizance of all or most of the design phase from the standpoint of their respective organizations.

The case study approach allowed a combination of data collection methods. A questionnaire was designed to capture insight into the primary variables that were expected to be relevant to the research questions. In addition to being a potential data source for analysis, the questionnaire was used to guide follow-on semi-structured interviews with each stakeholder. Program documentation provided insight into the adaptive measures enacted by each program, with preliminary observations being screened and validated in follow-on interviews.

The primary thrust of analysis made possible by the case study method was the detection of patterns in the programs. Once a means was established to differentiate between the programs in terms of their level of adaptability, it was possible to look for common characteristics of adaptable programs that were not present in less adaptable ones. Findings were formulated and related to previously developed theoretical constructs. This analysis is presented in chapter 6.

Another analytical focus was on the roles of the three primary stakeholders in making adaptability possible within programmatic constraints. For each adaptive process that took place during the design phase, critical stakeholder roles were identified based on interview data. The essential stakeholder roles provided a balance between structure and flexibility necessary to poise the participants in a “zone of novelty”, enhancing adaptability without creating conditions of chaos. Chapter 7 provides this analysis.

Appendix B presents analysis of several program characteristics to determine their relevance to level of program adaptability. These characteristics, requirements uncertainty, research and development budget, and the duration of the design phase, are

defined in section 4.5.4. This analysis was relegated to an appendix because it did not produce a significant finding related to program adaptability.

4.3 Sampling Strategy

The strategy chosen to implement this research method was to identify a set of case studies of Air Force command and control (C2) systems that had diversity in selected characteristics. C2 programs were chosen because they experience an acute need to adapt that is driven by changing threats and rapidly evolving technology opportunities. The C2 sector is dominated by communications and computer technology, both of which are changing at a rapid pace. C2 programs are driven to pursue evolutionary acquisition strategies and spiral development processes as means of combating technology obsolescence and responding to rapidly changing user needs.

Candidate programs were identified during exploratory research through interviews with program managers of 23 development programs at Electronic Systems Center (ESC), the Air Force's organization responsible for development of C2 systems. The selected programs had completed their design phase recently enough that interviews of key personnel were still feasible. Each program either involved replacement of an existing capability or had an acquisition strategy that incorporated elements of incremental deliveries and/or spiral development. This consideration ensured that all of the programs that were selected for study were in a position to incorporate learning from user operation of previous systems with related capabilities.

Programs were selected for diversity in their research and development (R&D) budgets, the duration of the design phase (design time), and the nature of the system representations used. Half of the programs were software developments, while the other half incorporated both hardware and software elements. Diversity among the eight programs was important to increase the likelihood that any impact of program variations on adaptability could be detected in the collected data. Table 4.1 lists programs A through H and the characteristics that were used to ensure selection of a diverse sample.

	Nature	R&D ($M)	Design (Months)	System Representation	Acquisition strategy
A	H/W & S/W	24	43	Development system	Incremental deliveries
B	S/W	23	15	Development software	Spiral development & incremental deliveries
C	H/W & S/W	13	14	Fielded prototypes	Follow-on to prototype system
D	S/W	22*	16	Development software	Spiral development & incremental deliveries
E	S/W	15*	6	Development software	Incremental deliveries
F	S/W	28*	21	Development software	Spiral development & incremental deliveries
G	H/W& S/W	140	24	Representative labs	Computer replacement
H	H/W & S/W	40	24	Representative labs	Follow-on to wartime contingency system

* Estimate – case study concerned a portion of the total development program

Table 4.1. Program characteristics

4.4 Data Collection

For each case, data were collected from three organizations: the SPO, a user representative, and the prime contractor. One or more individuals were selected from each stakeholder organization for interviews, although just one person was considered the

primary data source. Data collection involved questionnaires, interviews and program documentation. One questionnaire, from the primary data source, was used for each stakeholder organization in the analysis. However, interview comments were often collected from multiple individuals in an organization who had cognizance of the design phase and were available to participate in interviews.

The SPO individual selected for interviews was either the government program manager or, in the case of personnel turnover, a close advisor to the program manager who had knowledge of the design phase and was still present in the SPO. For six programs, the user representatives were individuals responsible for consolidating and validating user requirements at headquarters organizations. In one case, two user test representatives who were on-site at the contractor facility during design were interviewed. For the remaining program, an individual with experience from both the user headquarters and an operational field unit was selected. This person knew the program issues and participants even though he was not the individual responsible for program requirements. Contractor representatives had been the program management leads for all or part of the design phase in all eight cases.

The number of people interviewed and the number of interview sessions are provided in Table 4.2. The information is broken out by stakeholder. “Contacts” refers to the number of people interviewed for each stakeholder organization, and “sessions” refers to the total number of interview sessions, either in person or on the phone, that were held per organization. The total interview time for the 8 programs was approximately 130 hours.

Program	SPO	User	Contractor	*Grand Totals*
A - contacts	2	1	1	
- sessions	5	2	2	
B - contacts	3	2	1	
- sessions	6	2	2	
C - contacts	2	1	1	
- sessions	5	2	2	
D - contacts	2	1	1	
- sessions	6	2	2	
E - contacts	3	1	1	
- sessions	6	2	2	
F - contacts	3	1	1	
- sessions	6	2	2	
G - contacts	2	2	2	
- sessions	7	1	1	
H - contacts	2	1	1	
- sessions	6	1	1	
Total contacts	*19*	*10*	*9*	*38*
Total sessions	*47*	*14*	*14*	*75*

Table 4.2. Number of contacts and interview sessions

Table 4.3 lays out the sequence of steps that was taken to collect data for each program. The SPO collection activities were completed first to provide a foundational understanding of the basic information about the program and to validate the list of program adaptations (referred to as collaborative changes) before interviews with the user and contractor.

Activity	Description of Activity - SPO	Time (hours for typical case)
Introductory discussion	Introduction – describe research, get background information on program, and provide questionnaire.	1
Interview	Go over questionnaire, perform semi-structured interview, and identify program documentation, user subject matter expert (SME) and contractor SME.	1.5
Documentation review	Review of program documentation – identify potential collaborative changes (CCs). Note: some documents procured from other sources.	20 – 30 hrs.
Interview	Determine the list of validated CCs and evaluate CCs for value, cost risk and schedule risk.	2 - 4
Interview	Discuss collaborative highlights in depth – e.g. how SR was used, how stakeholders collaborated, powerful barriers or enablers, program challenges that were resolved collaboratively, etc.	1 - 2
	Description of Activity – User/Contractor (same for both)	
Introductory discussion	Introduction – describe research, work logistics of future interviews, and describe questionnaire.	.5
Interview	Go over questionnaire and perform semi-structured interview.	2 - 3
Interview	Evaluate CCs for value, cost risk and schedule risk, determine if any CCs were not on list, and ask if any CCs on list seemed invalid.	1
Interview	Discuss collaborative highlights in depth.	1 - 2

Table 4.3. Data collection sequence

The questionnaire (see appendix A) was given to each stakeholder after the introductory discussion and before the first interview. Answers to the questionnaire were discussed at the interview to provide opportunities for clarification or elaboration. Also, questionnaire responses often helped direct discussions in the interviews. The primary sections of the questionnaire were:

- Introductory material - research description and instructions
- Personal information
- Program characteristics (design phase, uncertainty, SR usage)

- Listing of stakeholder participants (user, SPO and contractor)
- Stakeholder knowledge sharing
- Stakeholder knowledge contributions
- Collaborative communication
- Stakeholder involvement in decision-making
- Knowledge transfer factors assisting and inhibiting knowledge transfer

The first interview was semi-structured in the sense that a list of questions (see appendix B) was prepared in advance. The questions asked depended on how each interview progressed, with a focus on aspects that seemed to provide the most insight into collaboration and adaptability. When possible, the answers to these questions were studied before the final interview to allow pertinent follow-up questions to be asked. Sometimes the first and final interviews had to be combined into one session.

The list of prepared questions for the interview followed the same topics as the questionnaire, but went into greater detail. Key questions included the following:

- **Describe the more significant uncertainties facing the program at the start of the design phase.**
- **Describe the SR in detail – what does/did it consist of?**
- **How were comments recorded and dispositioned? Describe the records available.** (Note: this question was used, primarily with the SPO, to help identify documents for the program documentation review.)
- **Define user POC for interviews. Discuss contact protocol.** (Note: SPO question only - this information was used to identify user interviewees.)
- **Define contractor POC for interviews. Discuss contact protocol.** (Note: SPO question only - this information was used to identify contractor interviewees.)
- **Describe how user/SPO/contractor unique knowledge was incorporated in the SR.**
- **Was there a significant program challenge (technology, funding cut, requirements change, ops concept change) that required knowledge transformation? Was the SR used?**
- **How was your organization able to generate knowledge for SR (experience, internal SR/database, collaboration with other stakeholders, etc)?**
- **Most important overall enablers/barriers of knowledge sharing.**

Program documentation was reviewed after the initial SPO interview and before other stakeholder interviews. Most documentation was located in the SPO, but in many cases user, contractor and test organizations were contacted to obtain additional records. Table 4.4 lists the documents that were evaluated for the eight programs.

The purpose of the document review was to make an extensive search for potential "collaborative changes," which were used as indicators of program adaptability. Most of the potential changes were screened out in later interviews with the SPO subject matter expert (SME) if they concerned anomalies, were never implemented, or otherwise did not meet the criteria defined in Table 4.5. Any program change, whether to requirements or to contractor design choices, that was implemented through the collaboration of two or more stakeholders and met the criteria was validated as a collaborative change . The number of candidate changes that were identified for each program is included in Table 4.4, and the number of validated collaborative changes is presented in the chapter 5 case study write-ups.

PROGRAM A (215 candidate changes)	**PROGRAM F** (99 candidate changes)
Configuration control board minutes	User survey feedback (briefing)
Monthly status reports	User's conference action items
Program development plan change summary	User's conference minutes
	Program status briefings
	S/W system specification
PROGRAM B (233 candidate changes)	List of program "add-in's"
Pre-configuration control board lists	
Software design description gov't comments	
Software req'ts specification – gov't comments	**PROGRAM G** (54 candidate changes)
Monthly status reports	Integrated Product Team (IPT) meeting minutes
Action items	Management review minutes
Concept of operations document	Preliminary design review (PDR) briefing charts
Program demonstration log	PDR meeting minutes
Program reports	Critical design review (CDR) briefing charts
Change log (formal exercise)	CDR minutes
Test issue matrix	Action item lists
Anomaly log	Requirements review board minutes
	Program issue lists
	Data management plan
PROGRAM C (77 candidate changes)	Integrated master plan
Change log	Hardware change decision briefings
Critical Design Review (CDR) issues list	Acquisition strategy panel minutes
CDR action items	Technical interchange meeting minutes
Program manager archived emails	Software development plan
Working group minutes	Architecture IPT briefing
Contract change letters	Specification change lists
System requirements document	
	PROGRAM H (110 candidate changes)
PROGRAM D (72 candidate changes)	Technical requirements document
Configuration control board minutes	Program manager's archived email
Working group minutes	Risk description document
Engineering change proposals	Baseline briefings
Task order description documents	Preliminary design review (PDR) briefing charts
Test case documents	PDR meeting minutes
Requirements document	Critical design review (CDR) briefing charts
User review minutes	CDR minutes
Working group meeting briefing charts	Decision briefings
Specifications	Program Executive Officer status briefings
Working group meeting agendas	Program change decision briefing
Pilot exercise reports	Technical interchange meeting minutes
	Affordability issues list
	User working group minutes
PROGRAM E (50 candidate changes)	User letter – required program changes
Test problem reports (454 reports)	Issues list
Baseline change requests (60 requests)	Integrated Master Plan
	Test plan working group minutes

Table 4.4. Program documentation

Collaborative changes <u>are</u>:

- New or changed requirements arrived at through collaborative interactions of two or more primary stakeholders (user, SPO, contractor)
- New or changed design choices arrived at through collaborative interactions of two or more primary stakeholders (user, SPO, contractor)

Collaborative changes <u>are not</u>:

- Fixing anomalies or bugs
- Internal contractor design trades, technology decisions, or implementation details
- Changes driven externally from the primary stakeholders (e.g. Office of the Secretary of Defense or Congress)
- Clarifications of existing requirements
- Related to development tools
- Changes accepted after initial stakeholder feedback (i.e. with no further collaboration to define or evaluate the change)
- Changes that will not be implemented until a future deliverable increment of the system (i.e. beyond the increment currently in the design phase)

Table 4.5. Criteria for collaborative changes

After the validated list of collaborative changes was developed, each stakeholder was interviewed to get a subjective assessment of the value, cost risk and technical risk associated with each change. The interviewees were also asked to identify whether each change was a requirements change or a design change. In some cases, user and contractor personnel added to or challenged the list of collaborative changes, and their position was reconciled through more detailed discussions with the SPO.

The final interview was sometimes combined with the first interview due to time constraints. In the final session, loose ends were cleaned up and information of particular interest from the earlier interview was explored at greater depth. Examples of these special interest items included:

- How did the SR and collaboration help create some new, significant capability during the design phase?
- How did the SR and collaboration help overcome a challenge (funding cut, requirements change, etc.)?
- Were there any special ways that stakeholders improved their knowledge and injected it into the design process?
- Were there any particularly powerful or unique barriers or enablers to collaboration?

Several challenges came up during data collection. It was sometimes difficult to locate program documentation, particularly when it was not available in the SPO. Contractor, user and test personnel had to be contacted in several cases to fill in missing documents. Another difficulty was that, given the large number of potential changes identified for some programs (as many as 233), it was hard to get enough time with the SPO SME to validate the list of collaborative changes. This process typically had to be done over two or three separate sessions. Lastly, identifying user and contractor experts and setting up the logistics of interviews with them was often an extended process. Running several case studies simultaneously helped alleviate this difficulty.

The assembled data is consolidated into case study write-ups for each program, which are presented in chapter 5.

4.5 Variables

The following sections provide definitions and descriptions of the relevance of variables that are part of the analysis effort presented in chapters 6 and 7. As indicated in Table 4.6, one dependent variable and six independent variables were defined.

Dependent Variable: • Program Adaptability **Independent Variables:** • Level of knowledge sharing • System representation fidelity (level of detail) • System representation fidelity (coverage) • Requirements uncertainty • Program funding • Design phase duration

Table 4.6. List of variables

In addition to these variables, several other variables that were initially defined, but were not of use in the analysis, are described in section 4.5.5.

4.5.1 Program Adaptability

For purposes of this study, adapting refers to a decision to change a program requirement or to modify a currently envisioned design choice as a result of stakeholder collaboration. It is important to emphasize that unilateral (i.e. non-collaborative) decisions for change by government stakeholders can be problematic. To illustrate this concern, consider that if the contractor does not assess the cost and schedule implications of user or SPO-driven change requests, the implications for program cost and schedule constraints may not be well understood. A pattern of unilaterally driven change can induce chaotic elements such as cost growth and schedule delays into a program. Because of distribution of knowledge and responsibilities among the stakeholders, collaboration is critical to adaptability.

Program adaptability represented how much an individual program adapted during its design phase. This variable was tied to collaborative changes, which were described in section 4.4. Each program implemented collaborative changes, although the number and value of changes varied widely. The definition of program adaptability is as follows.

Definition: **Program <u>adaptability</u> refers to the demonstrated ability of a program to make changes to requirements or design decisions during the design phase.**

If a collaborative change involved a change to requirements, stakeholders were asked to compare the priority of the requirements change to the priority of existing requirements. For design changes they were given a subjective scale to describe the level of added capability provided by the change. The five-point scales for these evaluations are listed in Table 4.7. Since three stakeholders were queried on the value of the changes, the rating given to each collaborative change was determined by the average of the three responses. Average scores could therefore be: 1.0, 1.3, 1.7, 2.0, 2.3, 2.7, 3.0, 3.3, 3.7, 4.0, 4.3, 4.7 or 5.0. If the average score for a change was between 1.0 and 2.0, it was considered a “high value” collaborative change. Averages of 2.3 to 3.7 were “medium value”, and averages of 4.0 to 5.0 were “low value” changes. These value ratings are used in chapter 6, along with the quantity of changes implemented, to define a means of differentiating between the adaptability levels of the programs.

Collaborative Change Evaluations

Requirements Changes	
Score	**Criteria**
1	**Top 20%**
2	**Top 40%**
3	**Top 60%**
4	**Top 80%**
5	**Bottom 20%**

Relative priority of change compared to existing requirements

Design Changes	
Score	**Criteria**
1	**Exceptional**
2	**Substantial**
3	**Moderate**
4	**Minimal**
5	**None**

Added capability provided by change

Collaborative change rating (value of change)	
High	**Avg. score 1.0 – 2.0 (n = 37)**
Medium	**Avg. score 2.3 – 3.7 (n=58)**
Low	**Avg. score 4.0 – 5.0 (n=37)**
	(total n=132)

Note: possible scores (avg. of three responses on five-point scale) were 1.0, 1.3, 1.7, 2.0, 2.3, 2.7, 3.0, 3.3, 3.7, 4.0, 4.3, 4.7 or 5.0.

Table 4.7. Collaborative change evaluations

4.5.2 Level of Knowledge Sharing

Knowledge sharing was the primary form of interaction between stakeholders that was observed during the cases. For most programs, some of this interaction involved usage of the SR. Interview data provided indications of the degree of knowledge sharing using system representations that was experienced by the program. These indications

involved both the depth and the frequency of interaction of stakeholders with the SR. Five levels of knowledge sharing were defined such that the levels could be determined based on interview data. These levels were: exceptional, strong, moderate, weak and none. The definition of knowledge sharing used in this research is provided below.

***Definition:* <u>knowledge sharing</u> refers to the use of a system representation to facilitate information exchange between stakeholders when this interaction pertains to the development of a new system.**

4.5.3 SR Fidelity

Stakeholder descriptions of the system representations were evaluated to determine salient characteristics of the SRs that might relate to adaptability. The primary characteristic that emerged was SR fidelity, which can be thought of as realism in reflecting the system design. Fidelity was broken down into two variables. First, level of detail referred to the degree to which the SR captured the full design. SRs were characterized as providing system, subsystem or minimal levels of detail. Second, coverage referred to the degree to which the SR captured emphasis areas of the government stakeholders. Coverage was defined as being either high or low. Government stakeholders identified emphasis areas for each program during interviews. These areas were program aspects that were considered to be of major importance.

***Definition:* SR <u>fidelity</u> consists of two elements:**

- **<u>Level of detail</u> refers to the degree to which the SR reflected the system design.**
- **<u>Coverage</u> of stakeholder emphasis area refers to the degree to which the SR portrays stakeholder emphasis areas.**

4.5.4 Program Characteristics - Requirements Uncertainty, Program Funding and Design Phase Duration

Several program characteristics were identified that might have an impact on adaptability. Data on the level of uncertainty in the requirements baseline were collected in the questionnaire and discussed with the stakeholders. Since questionnaire data was not reliable due to variability between stakeholder responses, a subjective assessment of uncertainty was done for each program based on stakeholder interviews.

Definition: **<u>Requirements uncertainty</u> describes the perceived degree of stability of the program requirements baseline at the start of the design phase.**

Data on each program's funding level were gathered during the SPO interviews. The SPO subject matter expert (SME) was asked to provide an estimate of the funding associated with the design portion of the program.

Definition: **<u>program funding level</u> refers to the amount of funding authorized for a development program for purposes of designing a system.**

The SPO SME was asked in the questionnaire to define the start and end of the design phase. The start of design was associated with completion of the formal requirements definition effort. The completion of design was associated either with the final, formal design review (typically called a critical design review) or with closure of any major actions from the final review. The design phase duration was the number of months between the start and end of the design effort.

Definition: design phase duration refers to the number of months between the start of design (associated with completion of formal requirements definition) and the closure of the final design review milestone.

These three variables provide potential alternate explanation for levels of program adaptability. Are programs with greater uncertainty more likely to adapt? Are programs with more funding or longer design phases more adaptable? Analysis of these questions is contained in Appendix B.

4.5.5 Other Variables

Several other variable were considered due to their potential impact on adaptability. Data were collected on these variables in the questionnaire, but analysis did not reveal any significant findings. In many cases, the data sample was too small or inconsistencies between different stakeholder perspectives made the data suspect. These variables are listed in Table 4.8. The table also includes a description of the causal mechanisms that were anticipated for each variable.

Variable	**Anticipated Causal Mechanism**
SR usage (timing)	Usage of the SR starting early in the design phase
SR usage (number of iterations)	Multiple exposures of stakeholders to the SR
Knowledge contribution (user, SPO, contractor)	Patterns of contributions by particular stakeholders in specific knowledge areas (e.g. operational considerations from users, cost constraints from SPO)
Communication (frequency)	More frequent communication
Communication (value)	Greater perceived value of particular communication mechanisms; number of highly valued mechanisms
Communication (delay)	Shorter time span between requests for information and responses from other stakeholders
Consensus	Degree of involvement of all stakeholders in decision making about design and requirements changes
Importance of factors assisting or inhibiting knowledge sharing	Recurrence of certain factors (e.g. user conflicting priorities, SPO staffing levels, etc.)

Table 4.8. Other variables

4.6 Roles of Stakeholders

One of the research questions defined in chapter 3 concerned the roles of stakeholders in making the design of a new weapon system a more adaptable process. This section provides a definition of stakeholder roles, discusses their significance for adaptability, and gives a description of the analytical approach employed in this research.

Robbins (1998) defines a role as “a set of expected behavior patterns attributed to someone occupying a given position in a social unit.” This definition implies that, because an individual is in a certain position, he or she has responsibilities to perform certain functions, and that these responsibilities influence their behavior. The “set of

expected behavior" constitutes the role of the individual. In this research, the three stakeholders can be thought of as groups of individuals who are collectively responsible for performing certain functions. These responsibilities will influence their patterns of behavior, or roles. The definition of stakeholder roles used for this research is the following.

***Definition:* <u>Stakeholder roles</u> refer to a set of expected behavior patterns attributed to an organization or set of organizations carrying out defined functions in the development of a new weapon system for the Air Force.**

To apply this definition, the "defined functions" of the stakeholders must be understood. As described in chapter 2, the SPO has two primary functions during design: planning and management of the design effort and enforcement of constraints and rules. The user defines and prioritizes requirements and advocates funding. The contractor integrates requirements and satisfies government objectives and constraints. It is worthy of note that if adaptability is not considered, the function of the user is unclear during execution of the design phase. Requirements, priorities and funding are all prerequisites to initiation of design. Once adaptability is considered, the role of the user in design becomes significant.

Adaptability can be thought of as an additional function or set of functions, although existing structures that support program objectives and constraints remain imperative. The question becomes - what roles must the three stakeholders perform to promote the function(s) of adaptability and to ensure innovation does not come at the price of loss of control? The analysis of stakeholder roles is described below and covered in detail in chapter 7.

In section 4.5.1, the point was made that collaboration between the stakeholders is critical to adaptability. With this consideration in mind, and as a precursor to assessing stakeholder roles for adaptability, a set of collaborative functions were defined that were observed to promote adaptability during the design phase. Once the adaptive functions were defined, specific stakeholder roles that supported these functions were delineated. These roles were best practices for facilitating adaptability that were identified and described during case study interviews with highly adaptive programs.

Some of these roles involved providing flexibility so that adaptability would be possible. Other roles involved structure to maintain control. The adaptive roles that were defined formed the nucleus for a set of recommendations concerning best practices for the stakeholders in support of adaptability.

4.7 Validity

During the design and execution of the research method, careful consideration was given to a set of potential validity concerns. These concerns and their implications are discussed in the following paragraphs.

Most of the data used for analysis were taken from interviews with the stakeholders. One validity consideration was the limited number of individuals that were contacted for discussions. Frequently, only one individual for each stakeholder organization was interviewed, which raised the possibility that lack of first-hand knowledge, incomplete memories or individual biases might influence responses. The most knowledgeable person available from each stakeholder organization was selected for discussions to help mitigate the concerns of knowledge and memory. Since most of

the interview questions concerned aspects of collaboration, it was possible to cross check many comments by comparing the responses of the SPO, user and contractor. This approach helped to minimize bias. In most cases, a coherent story emerged from the combined set of interviews. In instances where discrepancies were observed, they were noted explicitly in case study write-ups.

Questionnaire data were, for the most part, not considered sufficiently reliable to use in analysis due to the small sample size and large variations between the answers of the different stakeholders. When subjective data on requirements uncertainty were evaluated, these concerns were explicitly addressed in the analysis.

Another subjective measure that was used was the stakeholder perception of the value of the collaborative changes. These data were collected from all three stakeholders using a five-point scale, so each change had a potential variability between stakeholder responses. Averaging the variability values for all of a program's changes provided a sense of how different the stakeholder responses were for that program. These averages ranged from 0.4 to 2.0 out of a possible maximum variation of 4 (i.e. a case in which one stakeholder answered 1 while at least one other stakeholder answered 5). While this level of variability indicated that stakeholders did not share identical views on the value of the changes, the averaging of the three responses alleviated this concern. For 130 of the 132 collaborative changes, at least two of the stakeholders had identical answers or responded within one increment of response of each other on the five-point scale. Also, as explained in Table 4.6, the collaborative changes were categorized for analysis as high, medium or low, rather than being valued by the average stakeholder responses.

Another potential concern was that the review of program documentation might have missed one or more of the program's collaborative changes. This concern was addressed by conducting an extensive review of all the relevant program literature that could be located and listing any item that remotely resembled a potential collaborative change. The list of reviewed program documentation is provided in Table 4.4. In all, 910 potential changes were identified through the documentation review, of which 132 were validated as collaborative changes by the SPO using the collaborative change criteria. The user and contractor subject matter experts were then asked to evaluate the SPO list of collaborative changes, resulting in a total across the eight programs of eight added collaborative changes and ten deleted ones. The three stakeholders were asked to review these corrections until they reached consensus. The other primary safeguard was the nature of collaborative changes, which, by definition, took collaboration to evaluate. Such issues tended to be open for a period of time, which heightened the probability that they would be captured in program documentation.

The last area of data collection to raise validity concerns was the phenomenon of variability over time. Interviewees sometimes indicated that one answer to a question was correct for one phase of the program, and a different answer was appropriate for a different phase. In these cases, the case studies included text that explicitly identified the time-varying nature of the response, as described in the interview data.

4.8 Summary of Research Method

This discussion of the research method has described the case study approach that was used, including the sampling strategy and how the data was collected. A detailed description of the variables was provided, along with an introduction to the analytical methods employed in later chapters. The closing discussion of validity outlined some of the considerations associated with data collection and how potential issues were evaluated and mitigated.

The next chapter provides the case study write-ups for programs A through H.

Chapter 5: Case Studies

5.1 Introduction

This chapter consists of eight sections, presenting detailed descriptions of the eight case studies that were performed. Each study follows the same format consisting of sections on the following topics:

- Main body – provides program description and covers unique aspects of the program and of the three primary stakeholders (e.g. sources of complexity or uncertainty, co-location of stakeholders, special stakeholder knowledge or capabilities, number of users, etc.)
- System Representation Description – describes the nature of the program SR
- System Representation Usage – explains how the SR was used on the program
- Stakeholder Interaction – covers stakeholder roles and how the parties collaborated with each other to share information and manage the program
- Adaptability – summarizes the collaborative changes and describes the level of adaptability for the program
- Summary observations– recaps case highlights and provides lessons learned

The data from these studies is summarized in a matrix in Table 5.10 at the end of the chapter. The cases represent the primary source of data for the analysis that follows in chapters 6 and 7.

5.2 Case A

Case A involved a $24 million effort that took place over 43 months to design and produce a system through four incremental deliveries. The increments evolved during the development, adding greatly to the capability that was originally envisioned. The system, made up of both hardware and software components, provided an analytical capability that produced data needed by operational squadrons worldwide. One interviewee referred to the capability as a “back office system” since it was not operated directly by field personnel, but rather was used at a central location to produce data for field units. The Air Force System Program Office (SPO) received requirements from a single user headquarters, referred to here as Agency A. This Air Force agency operated the system and disseminated data to field users. The SPO had a contractual relationship with Contractor A to design the system, code and integrate software, and procure the necessary hardware.

Three aspects of the program had particularly important influence on stakeholder interaction in support of system adaptability. First, the users (Agency A headquarters personnel), the contractor and the system representation (SR) were co-located at Agency A’s facility. Second, the SR was used to generate data in support of field operations, which made user feedback available during development. Third, Agency A had a high level of technical expertise in the science underlying the analytical capability of the system.

Co-location in the same building facilitated daily, face-to-face contact between user and contractor staffs. User personnel were very focused on what they needed from the system, and, due to frequent discussions, the contractor indicated they had a good

idea what the customer wanted at any given time. As an example, the contractor understood a particular user difficulty – the need to get data to the field faster. Contractor A responded with an unsolicited adaptation initiative that resolved the user concern.

Operating the SR to produce a product for the field community during development led to the opportunity for feedback on user needs, which helped focus the stakeholders on opportunities for improvement. A senior manager in Agency A traveled frequently to gather field user inputs. He would then task his staff to investigate potential means of adapting the system in response.

The technical expertise of the user in the underlying science and modeling associated with the system enabled them to challenge the contractor to achieve greater levels of fidelity in their analytical algorithms. User personnel had a background and experience base in modeling, allowing them to work one-on-one with the contractor's programmers to assess the scientific reasonableness and relative utility of design options. The contractor indicated that the user's understanding of algorithms helped the contractor make decisions regarding the "low level implementation" of the design.

Because of its analytical nature, the system's performance was closely related to computing power. System adaptation was therefore fueled in part by the availability of constantly improving computer technology. Another factor in adaptation was the high priority placed by the user on keeping the system up and running to produce operational data for the field. Computational capacity and reliability were therefore strong considerations whenever potential adaptations were being considered.

All three stakeholders indicated they were able to maintain a clear, shared view of what capabilities the system needed to have, even though technology advances and

evolving user needs drove dramatic shifts in this view over time. User representatives attended code reviews, configuration control boards, and weekly Integrated Product Team (IPT) meetings, and they also participated in frequent, informal hall conversations. The contractor indicated the user had evolving operational needs, and they would sometimes have to prod the user community to define priorities. For the last two incremental deliveries, a priority list was defined explicitly and was collaboratively maintained with the user. From the SPOs perspective, the weekly teleconference helped the parties to maintain a shared understanding of what was going on and where the program was going. The user indicated their senior management was proactive in cleaning up any conflicts internal to the user community so there would be no confusion regarding priorities.

Several other factors were notable about this program. It had very stable funding, which, according to the user was “a large influence on success.” Both the user and the contractor were able to leverage the work of organizations in the private sector such as universities and independent research groups who were collaborating to accelerate advances in analytical algorithms. Also, the U.S. military’s contingency operation in Kosovo resulted in additional requirements and supplemental funding.

This program was successful at remaining within programmatic targets for cost and schedule performance while dramatically improving upon original technical performance specifications.

SR Description

The SR for this program consisted of the development hardware and software suite that became the deliverable system for the program. The contract included four incremental delivery phases, each of which became a new operational baseline. During the first of these phases, a government-provided prototype was used to generate data for the field. Because of this approach, the contractor, user and SPO had the benefit of an operationally functioning system to use as a SR within six months after the start of the design period. Each additional phase added more capability to the system, although the SR also underwent improvements in between formal incremental deliveries.

For the iterations following the government prototype, the contractor used commercial off the shelf (COTS) hardware and was provided a substantial portion of the system's software by the government for integration with their developmental software. These factors reduced the risk and the design and test time required to obtain each new increment of operational capability. The contractor selected a flexible architecture that permitted easy scaling up of processing power, and allowed portions of the system to be shut down for updating or experimentation without perturbing operational requirements for field support. This flexibility was valuable in examining potential adaptations as well as in modifying the system.

SR Usage

Contractor A and Agency A interacted with the SR extensively on a daily basis, with the latter operating the system once the first phase was completed. The SPO was indirectly involved with the SR, receiving information primarily from Contractor A on an as-needed basis.

Because of the operational use of the SR, the user and the contractor knew what tasks were running at any given time on the system. They could construct a scenario for modification of the system, determine what the impact would be, assess the added benefits, and decide how to minimize down time during implementation. This approach provided a high fidelity means to investigate potential system adaptations, saving time and reducing uncertainty and friction between the stakeholders. In one example, the community found it much easier to evaluate a new approach to processing inputs because of their ability to run the SR with the new algorithm and compare the results with real-world conditions. The high level of fidelity of this evaluation would have been impossible without the SR.

Another major benefit to having the SR from such an early stage was added experience in hardware integration. This experience reduced risk and sped up integration when new versions of hardware were added to the system.

In one example of SR usage, the program obtained a new processor from an Original Equipment Manufacturer (OEM) to evaluate with the system. The contractor worked with the user to determine what kind of tests would be needed to determine if the hardware would meet requirements and be of sufficient benefit to justify the cost of

procurement. The user helped set up and execute the tests. The SPO gave permission to Contractor A to spend time to do the testing. After the contractor characterized the performance of the new hardware, the user was able to justify a request for additional funds. The SPO acted quickly to add work scope and funding to the contract. All three stakeholders played a role in evaluating and implementing this collaborative change.

The program used the SR to facilitate adaptations in several significant ways. The SR provided the contractor and user with detailed knowledge of current system operations, which effectively established a shared understanding of the current state of the system design. This understanding helped the stakeholders visualize the work effort required, the benefit and the impact of potential changes. The contractor, with the SPO's knowledge and support, used the SR to experiment with potential adaptations, immensely aiding the ability to evaluate such changes and reach decisions on whether they should be undertaken. Implementation of approved changes was easier due to past experience with the SR. In summary, the SR allowed faster and more knowledge-based shaping, evaluation and implementation of potential adaptations to the system.

Stakeholder Interactions

The SPO's management philosophy was to stay out of the way of contractor progress on the design, and also to allow substantive daily interaction between the user and contractor. Both contractor and user interviewees were appreciative of the SPO's methods in this regard, which facilitated rapid progress. The SPO also worked programmatic issues such as contract changes or funding issues in a timely fashion. The

user indicated that the SPO was able to save them from bad choices in several cases when seemingly attractive changes might have resulted in program delay or cost growth. When issues arose in the community, the SPO would act as the arbiter to push for a mutually acceptable solution. SPO and user trust in the contractor was high, and grew stronger over the course of the program as the contractor continued to deliver substantive enhancements to system capability without encountering cost or schedule difficulties. SPO continuity was maintained due to a dedicated support contractor who could tell new users about the intent of past agreements.

Agency A typically didn't require multiple levels of approval when changes were being considered. At weekly meetings, agency representatives set priorities for upcoming efforts or changes, based on user needs and contractor recommendations. The importance of establishing and clearly communicating priorities was emphasized at the user's senior management level. This input allowed the contractor to work through a list of tasks until available funding was used up. Other tasks were dropped or delayed. Agency A maintained the system software once it was declared operational, sharing hard drives, drawings, diagrams, and documentation with the contractor.

Contractor A took great care to capture system changes, exceeding contractual documentation requirements. They were responsible for configuration management of system software during development. The contractor's proficiency was in software design, and they leveraged the user's science expertise to assess the quality of their design approach. User personnel applied their knowledge of the underlying science and algorithms associated with the contractor's work to assess and critique design options, ensuring a high level of fidelity in the system's analytical modeling capability. The

driving factors influencing the design for Contractor A, in addition to the original contract requirements, were user input (including frequent input from a senior manager), availability of new technology, and level of funding.

The contract itself was a very flexible time and materials vehicle, specifying the resources to be used by the contractor for the established scope of work. It was easy to add or subtract scope of work or funding. Contractor A indicated they looked on the effort as part of a long-term relationship rather than just one contract, so they were not driven by self-interest to push inferior solutions to the government.

The data produced by the system was in a single standard form for everyone in the field, which reduced the diversity of user perspectives and simplified consideration of potential adaptations. Having Agency A as the sole user representative meant that decisions on issues and opportunities could typically be made with internal agency coordination. For example, when Contractor A encountered technical issues, local user personnel could frequently answer questions immediately.

Potential system adaptations were identified in two ways. In some cases, Agency A's senior manager got ideas from the field, as described earlier. In other cases, user, SPO and contractor personnel came up with ideas for better ways to perform the mission. In either of these cases, the SPO would review the potential change to see if it fit in the scope of work of the contract. The program used a Configuration Control Sub Board (CCSB) to approve changes. The CCSB was effective because it was composed of individuals close to the work (as opposed to higher level managers) who shared a common focus and, according to one stakeholder, did not indulge in turf battles. The primary concerns were user priority, how much it would take to do the change and

whether the change would fit in the available budget. One stakeholder indicated that a key was empowerment – the CCSB had the necessary information and could make a decision to implement the change without taking time for higher-level reviews.

Frequent and open communication was another cornerstone of the program. The SPO shared the available budget to the contractor, which helped them plan program execution and consider possible adaptations. Participants from all organizations acted professionally, and individuals were not criticized at meetings. Everyone kept the same end goal in mind – developing a quality system for the user. The program used a weekly telephone conference for information exchange. All parties were very responsive to urgent requests for information. Contractor A felt comfortable to indicate if an idea was bad, and they took pride in offering alternatives to solve problems.

While the program did not have a formal document delineating the roles of the different stakeholders, the interviewees felt the various participants were motivated by a shared sense of mission and an awareness of program milestones. Contractor A management indicated they had a sufficient span of control to allow a certain amount of exploration of potential improvements without unduly wasting resources.

The contractor indicated that user interaction could sometimes be a distraction, raising a concern that the level of interaction experienced on this program might not scale well to a larger program. Sometimes user personnel would visit the contractor in the middle of their work. While this user input frequently amounted to tweaking of the system design, in the aggregate it had the potential to contribute to instability. Contractor A indicated this concern was controlled because of trust and close communication. Contractor personnel knew they could tell the user to go away if they were busy. The

contractor program manager had a good relationship with his senior software and hardware people, so information on potential changes and user discussions tended to get to him before too many resources were consumed. In some cases, user personnel would go to individual programmers and they would "noodle around" for a couple days to determine the feasibility of a potential adaptation before the programmer would go to a senior person. Sometimes these interactions led to identification of valuable collaborative changes, while other times they did not pan out. The SPO philosophy was that the contractor's management reserve was intended to support this type of effort. The contractor indicated that if the SPO had been more rigid about this type of expenditure, it would have broken down valuable collaboration between the contractor and user.

Adaptability

The program documentation review and subsequent sessions with the SPO subject matter expert led to identification of 19 collaborative changes that were adopted during the design phase of the program. SPO, user and contractor representatives provided input on the value of these changes. The categorization scheme described in Table 4.6 was applied, leading to the results shown in Table 5.1.

Number of collaborative changes	19
High value changes	12
Medium value changes	6
Low value changes	1

Table 5.1. Collaborative changes for case A

Some of the more valuable collaborative changes included the following:

- Increased the number of analysis cases that could be run concurrently.
- Increased the time for which system output data remained valid.
- Created a means for the user to select a subset of desired data from the overall analysis to enable faster data delivery.
- Increased fidelity of analysis model and therefore of output data.
- Added an additional output data format to enable data to be used by the civilian sector as well as military users.
- Modified software to allow incremental delivery of data to field users rather than waiting for complete analysis runs. Enabled field users to get useful data significantly faster.
- Provided more detailed scaling of the data to the user.
- Approved new hardware processor upgrade, which significantly increased processing power of the system while maintaining reliability.

During analysis, collaborative changes were placed in one of seven potential categories. For Program A, thirteen of the nineteen identified collaborative changes were in the area of technical performance, one related to the user interface, two were interoperability changes, one was a maintenance-related change, and two enhanced reliability of the system.

Because of their co-location and their daily access to a high fidelity SR, the Program A user and contractor shared a detailed understanding of the design as it was defined at any point in time. They also were able to assess potential changes rapidly and

with a high degree of credibility to determine the value and implementation effort of the change. All three stakeholders were very interested in exploring technology advances or new ways of performing the mission, which set the stage for identification and evaluation of the large number of collaborative changes implemented on this program.

Within the pool of case studies observed for this research, program A was at the highest observed tier of adaptability. As described in section 6.2, Program A and one other program were considered very highly adaptable programs.

Summary

Program A was able to leverage a high fidelity SR to evaluate a large number of potential system adaptations very rapidly and with high precision, permitting timely, knowledge-based implementation decisions. The stakeholders adopted a conscious strategy to use the SR to facilitate their collaboration in an effort to support insertion of continually evolving technology and of better ways of performing the mission. User A had anticipated the likelihood of program changes from the beginning, and both the senior manager and the working level personnel were energetic instigators and supporters of adaptive improvements.

The typical sequence of identification and evaluation of adaptations went as follows:

- A field user, the agency, the SPO or the contractor would identify a potential adaptation, which could be a promising technology or a new way of doing the mission more effectively.

- In many cases the contractor and user would try out one or more ways of implementing a change using the SR to help settle on an approach, determine the value of the adaptation and characterize the difficulty associated with implementation.
- The stakeholders indicated they had frequent and open communication to aid in assessment of the potential change, aided in particular by co-location of the user and contractor and trusting relationships with the SPO.
- The user conveyed needs and priorities on a continuous basis to the contractor, which helped define the relative priority of the potential change.
- The stakeholders in the CCSB made decisions about potential changes without requiring time-consuming higher-level reviews.
- Finally, for approved changes, the SPO implemented the requisite modifications to the contract with minimal delay.

Every step in the change process took place rapidly and efficiently, contributing to the large number of changes that were implemented on the program.

The most valuable lessons learned from this program are described below:

- The strategy of having the user operate the developmental system meant it was possible to get feedback both from headquarters and field users during the design phase. This timely feedback allowed many adaptations to be shaped, evaluated and implemented during the incremental deliveries of the system. If the feedback had not been available until test or fielding, it would have been more expensive and time-consuming to make changes.

- The three stakeholders maintained a shared view of user priorities as they shifted over time. Actively managing the priorities allowed for more efficient use of resources and faster decisions on potential changes.
- The flexible architecture chosen by the contractor allowed easier scaling of hardware processing, and also facilitated experimentation with and modification of the system.
- The user and contractor used the SR to experiment with approaches to potential adaptations, which allowed assessments of the costs, benefits and potential alternate approaches. The resulting insights diminished tensions between the stakeholders and provided quality information for decision-making about potential changes.
- The SPO's hands-off management approach was possible due to an environment of trust with the contractor. The advantage for Contractor A was the ability to make faster decisions and interact freely with the user to understand their needs and priorities.
- The SR made integration of new hardware easier because of past experience.
- The flexibility to try new ideas with the SR (user and contractor) was balanced by contractor management attention to prevent excesses and a culture in which contractor personnel could tell users if they needed to get back to work.
- The SR enabled the user to justify additional funds because the contractor was able to demonstrate the value of a potential change.
- The stakeholders had a strong collaborative attitude because everyone shared the common goal of getting a quality system for the user.

- Using a flexible contract vehicle meant that work could be added or subtracted quickly, making it possible to implement numerous changes within the time available for design definition.

One issue raised by the contractor was scalability – Contractor A's program manager did not believe it would be possible to allow the same kind of close interaction between user technical personnel and programmers if the number of personnel had been much larger. He was able to keep in touch with these interactions and ensure that potential changes aligned with program objectives (with the help of the SPO as needed) and overall program progress was not being impeded. Also, contractor personnel were generally able to tell the users when they were pressed for time to get essential work done. As a final check-and-balance, the program was managed to a fixed budget, so the contractor took responsibility to plan ahead and stop work when funds were exhausted.

Two primary ingredients drove the high level of adaptation experienced by this program. First, all stakeholders, including field users with experience operating the system, had a conduit to identify potential changes for consideration. Second, the stakeholders had the ability to make rapid decisions fueled by quality information as supported by the SR and the described patterns of stakeholder interaction.

5.3 Case B

Program B involved development of four command and control-related software applications, and integration work to allow the applications to exchange data. The code was to be hosted on a separately procured software system, referred to here as the “host software.” The Air Force System Program Office (SPO) hired Contractor B to develop the system using a two-pass spiral process, culminating in a single delivery and demonstration of developmental (i.e. pre-operational) software. A future increment was planned to establish an operational capability. The research and development budget for the program was $23 million, and the design phase spanned fifteen months. The user headquarters (referred to as “User B”) coordinated involvement of user representatives (“user experts”) who had interest and expertise in each of the four applications.

Requirements for program B evolved through a series of meetings with user experts and User B. User B controlled this process. Four distinct user perspectives existed, since each prototype application had what one stakeholder called a “user cult.” These groups initially had no common view of how they visualized the collective system. The design phase forced interaction of the user communities. By the time preparations were complete for the system demonstration, the user experts had congealed considerably due to the patterns of interaction described in the SR usage section below.

The system representation (SR) for this program was the most current build of the software. Contractor B put on periodic demonstrations with the SR for the SPO, User B and the user experts to show the progress that had been made in increasing functionality and getting the four applications to interface with each other.

All three of the Program B stakeholders operated by building a consensus with each other through informal teleconferences and emails followed by more formal discussions at Technical Interchange Meetings (TIMs). The TIMs were used to iron out issues and reach agreement. The stakeholders indicated that they did not tend to have conflicts on issues. All three parties contributed ideas as the design evolved, and the normal course of action was to get community buy-in to make sure everyone would take ownership of decisions. According to the SPO, the typical pattern was to work through discussion of an issue conceptually rather than by interacting with the development software (i.e. the SR), although exposure to the SR initiated ideas for many discussions.

From the beginning, the SPO put a strong emphasis on user involvement in the design process. They felt that user clarification of the intent of written requirements was an ongoing process. The SPO made sure the user knew they were listened to and that their inputs made a difference, and they made a conscious effort to nurture the momentum of user participation.

The SPO SME felt that if they had not had significant user involvement, the user community would have rejected the system. As a result of user interaction, the system changed a great deal (see the adaptation section below). The SPO felt it was necessary to have something that people could see and interact with in order to reach a shared understanding and initiate discussions. The SR provided this visual mechanism.

The major emphasis of both the SPO and the users during design was on the Graphical User Interface (GUI) of the software. Reviewing the GUI revealed what an operator of the system needed to do to achieve a task, what information was available, how it was shown, how the applications interacted together, and how long the system

took to do tasks. User feedback on these aspects of the design helped contractor engineers understand how an operator would actually use the system.

The SPO made extensive use of a federally funded support contractor in a systems engineering role. These personnel, co-located with the SPO, had a strong influence on software design and architecture in such areas as modularity, good design practices, etc. During interviews, the SPO subject matter expert (SME) emphasized that documentation reviews were a major source of interaction and influence over the software design.

According to the SPO SME, the primary technical challenge for the program was integration with the separately developed software that would host the Program B software. The SPO indicated they played a major role in tracking changes in the host software by coordinating with co-located counterparts who were managing the host software development. Maintaining this connection allowed the SPO to keep Contractor B in the loop in a timely fashion. The SR assisted in the integration process, as described in the SR usage section.

Contractor B highlighted that the lack of a stable concept of operations for the system was very significant for the contractor design team. As an example, when the Air Force was unable to define how many users would use the system, and what composition a crew would have (i.e. how many people would be using the different applications), it was difficult to do performance loading estimates to reflect usage levels. If the Air Force hadn't provided the information, the contractor would have needed to make assumptions, which the contractor SME felt was "dangerous." The advantage of having the SR was that the program could bring all the users together, facilitating the government's ability to

make these types of decisions. The contractor was able to use the resulting input to improve certain aspects of their design.

One important insight provided by the contractor was that contractual requirements to use an Earned Value Management System (EVMS) were difficult to mesh with a spiral development approach. True spiral development would include an evolving baseline, whereas EVMS forced the contractor to define a waterfall of activity, severely limiting future flexibility. This issue illustrates the difficulty inherent in combining deterministic command and control practices (in this case use of EVMS) with adaptability.

Program B had a large number of collaborative changes, and yet it remained within cost and schedule constraints.

SR Description

Case B's system representation (SR) consisted of the development software itself, as it existed at any given point in time. This software included versions of the four applications that could be run on separate workstations. The applications were partially integrated in the sense that they could exchange information with each other.

One of Contractor B's requirements was to host this software at a dedicated lab co-located with the SPO to replicate in-plant software. The SPO was therefore able to witness system demonstrations and test procedures without having to travel to the contractor plant.

For the user community and the SPO, the SR provided a means of interacting with the GUI of the system in a workstation environment.

SR Usage

The user experts found that looking at all four applications had an impact on their evolving expectations of what the system should do. By interacting with the SR and with each other, the user experts gained a mutual understanding and were able to envision not only the four separate applications, but also how they should be integrated. The SR also provided a means to force user convergence on holistic issues like crew sizing and function that were important to the contractor's ability to optimize their design. The SR was therefore a catalyst to facilitate user input, both on application integration issues and on how the system would be used by operators.

The SR also helped with the problem of integration with the dynamic host software baseline. The contractor received a copy of host software versions, and it became standard practice to run versions of the two software products together whenever new releases of either product became available.

The user community interacted with the SR in three different forums. The contractor held informal demonstrations in which users were able to interact with the SR. Also, users were given training in preparation for the final demonstration. Lastly, user personnel were the operators when the final demonstration was run.

The informal user interaction sessions were done as side sessions at design reviews and other meetings. Since the GUI was central to understanding how operators would use the system, it was the major emphasis area considered by User B and the user

experts. According to the user SME, the significance of hands-on interaction was best described by the saying that “ a picture is worth a thousand words.” Users could sit down at a workstation knowing what they wanted to be able to do, ask the system questions, and see how it responded. Discussions with contractor engineers could usually resolve issues in a short time. This ability to provide on the spot interpretations of their observations to the contractor was “at the apex of importance” for the user. They felt that, absent this means of conveying understanding of their needs, the program wouldn’t have succeeded. This input was valuable to the designers since it enabled them to match system characteristics more closely to user expectations, reducing the risk that issues might arise during testing.

The user experiences with these informal sessions highlighted an observation about the nature of the design process. The user SME felt that when the design was well executed, the system would be easy to use and the user could intuitively grasp how to get and display information. This user-friendly aspect of the system is almost impossible to define in the requirements phase, but it could be seen and evaluated through interaction with the SR during design.

If an application was new to the Air Force, the demonstrations had even greater value, because they could provide the users an opportunity to explore what they would do with the application and therefore determine design preferences that would not otherwise have been apparent due to lack of historical context. The user indicated they had not had the time or the means to think about these issues beforehand. The demonstrations were valuable as a means to visualize operation of the new application, rather than just relating the concept of its use.

Contractor B noted that, at meetings such as the periodic TIMs, user personnel did not seem to be interested in design-related discussions. He described using “object interface diagrams”, power point slides and design documentation to describe the design, and stated that these methods did not seem to elicit any user input. However, the users were highly engaged in the demonstrations, which allowed them to pull from their operational experience as they interacted with the system and observe how data was exchanged between applications. The visual nature of the SR and the ability to interact with it seemed to be a key enabler to pull the user into the design process.

The contractor SME felt that the demonstrations were started too late. Earlier user feedback would have been useful in making design choices. He suggested that once a functioning product (code running on a system) with the anticipated look and feel of the system was available, it would be productive to share it with users. The timing of demonstrations in Program B was driven by resource tradeoffs early in the program.

During the training sessions held in preparation for the final demonstration, the contractor provided instruction to 15 trainees on the four prototype tools. The last day of training involved running a scenario. User B personnel helped structure the scenario, including the inputs that would be given to the four applications. In order to reflect a real-world configuration, the SR was put on multiple workstations, each with a crewmember who would operate in one of several intended capacities. Operator call signs were established and a communication network was established between the stations. The operators were given a mission briefing with commander’s priorities, and the workstations were fed scripted inputs. Information was transferred automatically from machine to machine and summary messages were generated, as would happen in a

real-world application of the system. Adding the personnel and ancillary equipment considerations to the training exercise with the SR permitted a much greater level of realism. In this configuration, the SR provided an opportunity to check interfaces and interoperability between the applications. The training also led to considerable dialog about what the system should be able to do in the future, which was captured for further consideration.

One user stated that the scripted inputs were a limiting factor in the utility of the training since they were resource-intensive to produce and did not provide the fidelity to reflect a real-world situation. This observation highlights that the fidelity with which external interfaces are simulated is one potential limitation of SRs. As with the timing issue of performing demonstrations with the SR, this consideration was resource-related.

At the final demonstration, the user employed four workstations. Evaluators focused on a limited scope: system effectiveness, suitability for the operational environment, and interoperability with the host software. Several comments were generated for evaluation in conjunction with future increments of the system.

Stakeholder Interactions

The SPO held a series of informal meetings over an eight month period with the user to define where the majority of their focus lay, and in particular how they would like to see arrangement of displays. This amounted primarily to top-level feedback on the GUI. The SPO recalled having a hard time getting continuity of users for these sessions.

The program went through a requirements clarification phase using a series of teleconferences, meetings and TIMs. The stakeholders took each requirement and went into detailed discussions on what it meant operationally. Sometimes the SPO and contractor interpreted these discussions as additional requirements, but these issues were resolved through discussions. The user indicated they worked out a lot of those types of issues, and when necessary they deferred some work to future increments in order to live within financial limits of the program.

TIMs were held throughout the design phase, which allowed everyone to get together every six weeks. The TIMs served to keep everyone informed, and all participants were expected to communicate openly. The SPO did not claim veto authority on system decisions, preferring to build consensus of all three stakeholders. The SPO felt that expectations regarding what everyone should do were clear. People were encouraged to be forthcoming if they made mistakes or needed help, regardless of organizational boundaries. As individuals got to know one another they came to appreciate that they had a common objective to provide a useful system, and this helped all the parties work together harmoniously.

Many issues were worked through informal communications (email, phone, etc.) and then formally discussed and resolved at TIMs. In rare cases, teleconferences or meetings would involve just the SPO and contractor, but for the most part all three stakeholders worked together. A weekly teleconference was held with all program stakeholders, and an agenda was generated in advance. The contractor indicated this practice was valuable and helped manage risk by keeping all the stakeholders aligned.

The SPO felt the user knew what their involvement should be, although their actual participation was often a function of how busy they were. If money was short, the user had to establish priorities for the program. In several instances, user input would lead to discussion to reach resolution within a design trade space. Many of these discussions focused on how to improve functionality. One example was display filters, which determined what information out of the available set of data would be on the display. The user was active in collaborative discussions regarding how this aspect of the system should be implemented and what symbols should be used to show various kinds of data. One central user message was that maintenance, including such issues as training and documentation, needed to be factored in as long-term considerations.

The contractor indicated a concern with the level of user participation during design. As an example, very few users were present at the design reviews. He sensed that involvement in prototype integration was not a primary focus of the user community, although part of the issue seemed to be that the user HQ staffs were overextended. The contractor felt the interaction changed as preparations started for the final demonstration. He observed a much stronger team collaboration, more common focus, and greater user participation. The contractor mentioned that co-location (the contractor and user were in the same town) had huge potential, but this was largely an untapped value.

Another issue that emerged from the interviews was the perceived quality and adequacy of understanding of the system concept of operations. The user felt this information was developed and conveyed, and the contractor felt the absence of clarity in some areas regarding how the system would be operated was problematic for designers. As discussed in the summary section, this disconnect implies a need for greater shared

understanding between users and contractors regarding the importance and usage of a program concept of operation.

The SPO expended a great deal of effort to set the stage with the user community going into the final demonstration. The SPO, user and contractor participated in weekly meetings leading up to software installation at the user's facility. This process helped align the expectations of the parties and served to get logistical details resolved. The key document was the system specification, which set the requirements baseline for the system. One central tenet was to ensure the user understood that they should evaluate the system on the established baseline, and not on future expectations for the system.

Adaptability

This program was characterized by a large number of collaborative changes, and by the fact that most of the changes fit a similar pattern of value and risk. Stakeholders described seventy percent of the changes as having both low value and low risk to implement (see Table 4.6 for an explanation of the value scale for collaborative changes). All 34 collaborative changes for Case B were concerned with the user interface. By their nature, most of these GUI changes posed low cost and technical risk to implement while the software was still in the design phase.

Table 5.2 provides the total number of collaborative changes that were observed for the program, along with a breakout of how many were assessed to be high, medium and low value by the stakeholders.

Number of collaborative changes	34
High value changes	4
Medium value changes	6
Low value changes	24

Table 5.2. Collaborative changes for case B

The following are examples of GUI changes that were given a low value/low risk rating during stakeholder interviews:

- Defined ground rules for setting default probability values in certain circumstances.
- Incorporated altitude parameter into positional data, along with latitude and longitude information.
- Provided a means of querying the system and filtering a list of existing products.
- Added ability to duplicate objects on the screen and create objects from other objects.
- Incorporated the ability to edit a map object by selecting it on the map rather than going through a more laborious process to access the edit option.

The program executed other collaborative changes, including the following, which the stakeholders considered to be of high value:

- Provided an error notification when connection is lost between two of the applications.
- Ensured labels of certain types of data are consistent between applications.
- Identified that live feeds of certain data were necessary for one application.

Program B was at the highest observed tier of adaptability within the group of 8 cases studied for this research. As described in section 6.2, Program B and one other program were considered very highly adaptable programs.

Summary

The Program B stakeholders involved four elements of the user community in a combination of informal communications, formal meetings and interactions with the SR in order to incorporate the user's operational perspective in the system design. The user interaction emphasized feedback on the system GUI. All three stakeholders felt that ability to incorporate this user feedback was critical to program success.

Lessons Learned from Program B include the following:

- Using the SR with supporting discussions helps users with different cultures and perspectives to converge on their views of how the system should meet requirements.
- The SR helped the contractor keep up with the integration challenge posed by host software that was evolving over time.

- The SPO strategy of encouraging user participation made members of the user community aware of their importance to the design process and resulted in a higher level of user engagement.
- User requirements did not capture design implications such as what steps a user must go through or how long the system can take to perform a certain task. This gap between requirements and design detail meant that user interaction with a SR to understand design choices had an important place in the design process.
- Contractor B recalled that user personnel did not seem to provide feedback on the system design in response to meetings, briefings or documentation. Demonstrations with the SR, by contrast, resulted in substantive user feedback.
- The user found the SR to be valuable in assessing new applications that had no legacy system context to use as a comparison. The user appreciated the ability to see a potential design in order to form a sense of how they would like to be able to use it.
- Open communication and consensus-building at TIMs was cited by the SPO as essential to maintaining strong, trusting relationships between the stakeholders.
- Weekly teleconferences helped align stakeholders, reducing program risk.
- The SPO observed that the contractor became more comfortable with user collaboration after they realized they had a common objective of delivering a quality system. System considerations then transcended individual differences.
- The contractor and user had different perceptions of the adequacy of the concept of operations (conops) for the system. The user felt the conops was well conveyed, and the contractor wanted more details to take the guesswork out of some aspects of the design. This observation argues for a more explicit declaration of needs by the

contractor, framed in terms of the impact to design decisions, so that contractor and user perspectives can be aligned regarding what conops information is required.

- The SPO documented system requirements and coordinated with the user community to align expectations in preparation for a demonstration to the user. Coordination achieved buy-in and safeguarded trust in the SPO/user relationship.
- Deterministic mechanisms like Earned Value Management require detailed planning that is at odds with adaptability. This issue represents a policy challenge that should be assessed by the Department of Defense if their stated intent to make systems acquisition more agile and adaptable is to be pursued.

The following two aspects related to the potential effectiveness of SRs were raised during the case.

- The user mentioned that the fidelity of external interfaces, which frequently had to be simulated during exercises with the SR, had an impact on realism, potentially limiting knowledge of system design implications.
- The contractor indicated that a SR with the look and feel of the envisioned system could have been used earlier to gain valuable user feedback on the evolving design. An earlier start would give the user community more time to digest and evaluate the contractor's design, and would result in more timely feedback to the contractor.

Both of these issues were related to resource constraints. Involving all three stakeholders in discussions of these two issues has the potential to increase value of the SR if the parties are able to apply their unique program contexts (such as limits on

availability of user personnel, resource constraints, and the nature of the system being designed) to the thought process.

Program B demonstrated a strong interaction between the three stakeholders, characterized by open communication, consensus building and a shared sense of the importance of developing a quality product for the warfighter. While different views were perceived regarding the role the user should play in the design process, the stakeholders agreed that tremendous value was added to the program due to the user interaction that did take place. These interactions happened both through usage of the SR and through a supporting pattern of informal and formal interactions.

5.4 Case C

Program C was a hardware and software development to provide an added sensing function and updated message generation capability to an existing deployable system. The R&D budget was $13 million, and the design phase as defined for the case lasted for 14 months. A small number of prototypes (which this case treats as the system representation (SR) for the system) were in operations overseas. A user headquarters office (User C) consolidated requirements for the system. Contractor C had provided the prototype systems under a previous contract, and they were tasked under this effort to develop a more operationally suitable version of the system that would carry forward the prototype capabilities while adding greater data link compatibility and improved sustainability.

The SPO indicated that discussions during the contractor proposal phase initiated the design phase of the program, even though the contract had not yet been awarded. This case therefore defines the design phase of the program as the timeframe between the government release of a Request for Proposal (RFP) and closeout of the Critical Design Review (CDR) action items, two months after CDR.

The prototype history had a bearing on the user perspective going into the Program C development. The prototype strategy was to use a limited amount of funding to demonstrate a set of capabilities, rather than to develop an operational system for deployment. Logistical details such as sparing, documentation and training were not funded. However, field commanders ordered deployment of the prototypes, and contractor logistical support was established. The contractor subject matter expert (SME) indicated that technical issues existed at the time the prototypes were fielded, but that

without access to the units, it was difficult for the contractor to continue troubleshooting the issues. The operators of the prototypes were frustrated, both with the difficulty of supporting the system and limitations in the capabilities it provided.

The user perspective for this case study came from interviews with a user C representative, referred to as the user SME. User C had to foster communication across organizational boundaries in the user community (field units, Air Force major command staff offices, and their own office), which made it difficult to form an integrated picture of user needs and priorities. Field users knew what they wanted and, although they had no direct communication with user C, they contributed a "wish list" of requirements through their respective headquarters. These major command headquarters were involved in the prioritization exercises that defined requirements for the program. Both the other headquarters personnel and field personnel tended to view User C, who was the interface to the acquisition community, with suspicion for being too close to the SPO and the contractor. Another impediment to communications was that the field units distrusted the SPO and contractor, and they did not understand the constraints posed by limited funding.

The prototype system was used as a baseline for defining the requirements of the new system. In the pre-award phase, the stakeholders focused on defining changes from the existing requirements that were affordable. Contractor C's preliminary cost estimate in response to the government's original requirements was twice the available funding. The pre-award phase was extended while the SPO, the contractor and User C met to converge on a prioritized subset of requirements that would remain within fiscal limits.

Communication between the SPO, User C and the contractor was open and effective. The relationships involved direct interaction and a shared interest in defining an acceptable program for the available budget. The SPO, the contractor and User C worked closely together, communicating frequently and meeting at monthly technical interchange meetings, rather than exchanging documents and awaiting responses from the other parties. The government emphasized hard requirements and priorities, while the contractor provided feedback on costs. Many software capabilities that were originally requirements had to be weeded out. The requirements that survived this process included porting the prototype's software functions to new hardware and a new operating system, improved data link compatibility with other systems, upgrades to hardware due to obsolescence concerns, and sustainability aspects such as documentation and training. Some issues were deferred to achieve contract award before the program lost funding.

The SPO and the contractor received no direct feedback from the field units operating the prototypes, although the SPO was able to make use of corporate memory of co-located support contractors who had worked on the prototype development. In particular, these individuals understood what requirements had been verified earlier in testing and what issues remained open for consideration during Case C requirements definition, design and testing.

The SPO program manager had an operational background, having served in a deployed unit of the type that would be using System C in the field. All three stakeholders indicated that it was valuable to have an individual with this background. He intuitively understood user requirements, and could recognize when solutions did or

did not make sense. Also, he could identify problems from a user perspective as the design emerged.

The contractor had some continuity between the prototypes and System C, due to corporate memory regarding how the prototypes had been designed. Also, contractor personnel went to the operating locations to provide technical assistance, which helped them develop an understanding of how the users were operating the system. However, the prototypes were being used to varying degree of effectiveness at different sites, making it difficult to ascertain how best to use the system.

The program had no formal Concept of Operations, or conops, from the user community. All three stakeholders felt this was a problem for testers since it was hard to develop appropriate test plans without knowing how the system would be used. The SPO SME felt the lack of conops definition meant that some processes associated with operating the system were not planned out or were uncertain. He could not tell how much of a problem this might create, but he thought field users might have to figure out how to make the system work in some situations. If operators had been involved in design discussions, they could have provided insight into how they would use the system, which may have led to better design decisions.

Several funding-related factors played a role in making Program C less adaptable than many of the other cases. The shortage of program funding limited design flexibility, since many aspects of the design had to be locked in before contract award to ensure the system would be affordable. Contractor C was liable for all cost overruns under a fixed price contract arrangement, which made them risk averse. Any “what if” exercises would have come out of the contractor’s funds. Lastly, the affordable set of requirements put on

contract was a subset of user needs, so it would have been difficult to trade off requirements to make any new ideas affordable.

This program had the smallest number of collaborative changes of the cases studied, although the stakeholders considered several of the changes to be valuable (see the adaptability section). Program C was successful at meeting the cost and technical objectives of the contract, and it exceeded schedule objectives.

SR Description

The SPO SME felt that it would have been impossible to do Program C for the given amount of funds if not for the previous work done on the deployed prototypes. The prototype system acted as a system representation for the program, although in a limited sense. Since the SR was inaccessible to stakeholders, it provided a static representation of the new system during the design phase.

Limitations in the SR are described in this section. The SR usage section describes information regarding the prototype system and how it was used during System C's design.

The following limitations applied to the SR.

- The contractor and SPO had no access to the SR, which were controlled by theater commanders overseas. Therefore, knowledge provided by the SR was limited to historical insight.
- No direct mechanism existed to share knowledge that accumulated from operating the SR to the Program C designers.

- It was not possible to modify the SR to investigate the cost or benefit of design options.
- It was not possible to run scenarios with the SR to assess performance or troubleshoot anomalies.
- Field users from different regions gave divergent feedback on the SR, particularly regarding its functionality. This inconsistency made requirements definition and design more difficult.
- Changing data standard meant the SR no longer met interoperability requirements. If the SR had been accessible, it would have made development of software for the new data link standards easier.
- The SR did not fully represent the new system because hardware and software obsolescence issues, along with some additional functional requirements, made it necessary to re-accomplish significant portions of the design.

SR Usage

Historical information from the prototype development was the only benefit of the SR. The following list describes the information that was available to stakeholders and how it was used during System C design.

- Knowledge of the technical details of the prototypes provided the requirements baseline for a very detailed System C specification.
- Operational field users generated requirements lists based on their experience operating the prototype systems. These requirements often took the form of

problem areas that users hoped would be resolved in the new system. During the System C proposal phase, these problem areas were candidate requirements for the program. Priority and funding determined whether they were included.

- The contractor used the prototype design as an initial design baseline.
- Cost information for prototype hardware and software development helped the contractor to define and price design options for System C.
- The contractor had technical performance information on the prototype design, and was aware of technical problems that had arisen with the prototypes. This knowledge helped highlight issues to look out for during System C design.

Stakeholder Interactions

The most wide reaching collaborative effort during the design phase occurred during the period from the government's Request for Proposal (RFP) release to contract award. The SPO, the contractor, User C, and other user headquarters representatives participated in monthly Technical Interchange Meetings (TIMs) and frequent teleconferences to establish an affordable set of requirements for the program. No field user representatives were involved.

This activity happened over a period of months, extending the date of contract award. At the first TIM, one SME observed that user representatives did not seem to be familiar with the government specification. By the time of contract award, the specification had been worked line by line to identify issues. User concerns included manning levels to operate the system, training, environmental issues, and sustainment

issues such as parts obsolescence. The SPO worked closely with User C and the other user headquarters representatives to develop prioritized requirements. Then the SPO gave inputs to the contractor, who generated possible design approaches and cost figures to satisfy the requirements. User personnel were typically not involved in the design level of detail. The parties had to go through several iterations before the cost converged to the available budget.

Contractor C noted that the user representatives could have imposed requirements that would have led to failure, but a shared sense of the importance of delivering the system seemed to enhance trust and contribute to successful resolution. The user SME felt the rapport with the SPO was excellent during this period. Government action officers and support contractors did a lot of conferring either on the phone or in person. Tapping into SPO expertise helped the users know what they were buying. The SPO noted that user involvement dropped off, consisting of primarily the User C representative, after the minimum requirements were identified.

After contract award, the SPO was involved in the design process at arms length. Much of the design was already established due to earlier agreements. As issues arose, the SPO put information out to user organizations. Many of these issues were worked in teleconferences, emails and document reviews, although direct interaction at meetings was found to be more effective in generating a common understanding. The SPO commented that it was often difficult to get user representatives to meetings, including design reviews.

Throughout the design phase, the SPO found it helpful to have User C as a single point of contact for the user community. Keeping track of the large set of operational and

sustainment-related issues raised by different user headquarters organizations was difficult, and the SPO SME noted that User C did a good job of coordinating a single, consolidated user position. On some occasions, User C communicated directly to the contractor to get clarifying information. User C put pressure on other user organizations to get timely answers on issues. They typically had to give the other headquarters offices a straw man position to start working from, or they would hear nothing back. While individual action officers would give informal answers, getting formal organization positions was difficult and time-consuming.

One of the headquarters offices ("Ops") that fed requirements to User C was responsible for operational considerations. The Ops office was the conduit for communications with field units, and they traveled to the field sites where the prototypes were located. This office had several comments to the specification as it was being developed. Ops became the requirements integrator for several user offices, and they often had to resolve conflicting positions, even within the same Air Force command. When necessary, Ops personnel contacted the SPO directly for information.

During the proposal phase, User C had to report back to the other headquarters offices that many of their requirements were not affordable. Eventually User C had to sever the dialog and work with the SPO to get the effort on contract. The User SME indicated this was not ideal, because it was always better to get input from more people in the user community. In some cases, User C felt like it was necessary to make somewhat arbitrary decisions when user group consensus was not forthcoming. The imperative was to have a contract in place before funding was pulled.

During the proposal phase, the user did not pull in any field user experts for meetings. Headquarters personnel from the Ops office represented the field perspective. Once the effort was on contract, User C did get some field experts who had worked previously on the prototypes to attend some of the program reviews and the CDR. The user SME recalled that these individuals were very unhappy, since the SPO was not buying many of the things they wanted. They did not appreciate the impact of funding limitations.

Contractor communications with the SPO took place on a daily basis after contract award. Contractor C did not have much direct contact with members of the user community. The contractor SME observed that the user seemed to have many other priorities, and they were not as responsive to questions as the SPO. For some issues, it could take a month for the user to get back to the SPO with an answer.

The contractor developed four builds of the system software during the design phase. The SPO was involved in the verification process to ensure requirements were being met, but they did not focus on design aspects of the software. The contractor was in charge of making design choices, with the SPO having veto authority. One test agency, acting as user representatives, went to the plant to see the contractor's setup, although they were not major participants in the contractor's development effort. The software builds were not released for evaluation prior to formal testing.

The stakeholders were able to react quickly to resolve a couple of major design issues by evaluating and implementing collaborative changes. One issue involved the hosting of some of the system software on a processor that had obsolescence concerns. Contractor C developed cost information for an alternate design concept in which the

software would be re-hosted on an existing system processor, causing a small amount of performance loss. The user community was consulted, and agreed to accept the performance loss to achieve a more straightforward design and avoid a long-term obsolescence concern. Coordination and consensus between the three stakeholders took place quickly, avoiding the cost of delays or rework.

The user SME observed that there seemed to be holes in the SPO's situational awareness of other programs that could be related to System C. For example, the SPO was not following the upgrade of an aircraft that would be part of the message network along with System C. There seemed to be no process to track interoperability issues between systems unless the SPO happened to think of the connections.

The program had a joint Configuration Control Board (CCB) to review and approve formal documents. The User SME was very knowledgeable, and came to meetings as the user spokesperson. The SPO commented that this representation had a positive effect, especially since the user SME understood SPO cost constraints. The User SME faced a challenge consolidating perspectives from the different user headquarters, but this individual's ability to make decisions enabled the program to keep moving.

Adaptability

Table 5.3 provides the total number of collaborative changes that were observed for the program, along with a breakout of how many were high, medium and low value (as defined in Table 4.6.)

Number of collaborative changes	8
High value changes	5
Medium value changes	3
Low value changes	0

Table 5.3. Collaborative changes for case C

Of the collaborative changes on the program, one involved technical performance, two related to the user interface, one was for life cycle costs and two each related to reliability and development costs.

The collaborative changes for this program included the following:

- Defined minimum acceptable implementation of data link connectivity (development cost-driven change)
- Developed capability to differentiate between simulated and live sensor data
- Re-hosted some software, mitigating a concern with obsolescence of the previously envisioned host processor
- Developed alternate workstation concept (development savings)
- Provided notification to operators of link failure
- Added EMI shielding of crypto equipment
- Added EMI shielding of external connections

Compared within the body of case studies performed for this research, program C was considered moderately adaptable, falling in the third tier of adaptability. As

described in section 6.2, four of the cases exhibited more adaptability, one exhibited less adaptability and two other cases were also rated as moderately adaptable.

Summary

Program C completed the design phase of the program on cost and ahead of schedule, but funding limitations contributed to an implementation that was not fully satisfying to end users. Also, because of the lack of direct field user involvement, the SPO SME voiced a concern with whether the chosen design implementation might pose challenges for the end users.

Three aspects of the program limited stakeholder collaboration. First, the funding pressures described above constrained design trade spaces and made the contractor risk averse. Second, the inability of the stakeholders to interact with the prototypes limited their value as system representations. No dynamic representations of the evolving design were available to aid collaboration. Third, the number of diverse user organizations and the lack of field user involvement made it difficult to incorporate the user perspective. The contractor was given a prioritized set of requirements that could be satisfied with available funding, but they made design choices to satisfy these requirements with minimal user input. These factors may have contributed to the moderate level of adaptability experienced on the program.

The following is a list of lessons learned for the program.

- Fielding of prototypes without sustainment and before resolution of known technical problems puts a burden on the field, makes problem resolution difficult, and strains future relationships with the acquisition community.
- Strong participation and communication between the three stakeholders was essential to define an affordable set of requirements before contract award. Stakeholder knowledge (such as SPO understanding of funding constraints, user understanding of priorities and contractor understanding of the cost of options) had to be shared to reach a viable consensus.
- The following program constraints were factors inhibiting adaptability after contract award: limited funding, use of a fixed price contract, predetermined design, no funds for "what ifs", and lack of requirements that could be traded off.
- Because there was no CONOPS and no field user involvement in design discussions, the program was at risk that design decisions related to how the system will be used might not be optimal for the user.
- The SR contributed historical knowledge that helped the stakeholders, particularly during planning and establishment of requirements and design baselines. It also helped when the contractor was pricing design options. However, lack of access to the SR hindered the development effort.
- No method was in place for the SPO to watch out for interoperability issues that might impact the System C design. The SPO should be considering system-to-system integration issues to avoid or mitigate the need for redesign or modification efforts.
- User C acted as an integrator and bridge from the user community to the acquisition community. However, in doing so they were distrusted by the user side, which did

not have an understanding of acquisition issues and constraints. As a general observation, the stakeholder organizations tended to have more trust when they understood each other's constraints and methods.

- The operational experience of the SPO program manager, along with SPO and contractor corporate memory of the prototypes, helped provide the stakeholders more knowledge to support design choices and resolution of system issues.

The quality of stakeholder working-level relationships before contract award contributed to their ability to converge on an affordable design. However, constraints on collaboration after award may have limited the prospects for adaptability.

5.5 Case D

This case concerned one segment of a $90 million software design effort. System D provided Air Force operators a set of capabilities for logistical planning of asset deployment to the field. The new system replaced an existing or legacy system that was being operated by the Air Force. System D software was to be hosted on a previously developed Air Force software system (the "host system"), and was intended to pass information to a joint software system (the "joint system").

System D included four functional segments, each of which had a different user community. This study focused on one segment that added analytical and reporting capabilities. User D was responsible for defining requirements and representing user perspectives for this segment to the SPO. The user subject matter expert (SME) for this case had been with User D for many years, and had forged connections with the primary members of the user community in his functional area. User D was therefore in a strong position to integrate user perspectives.

Before contract award, the SPO worked closely with three contractors for a six-month period to generate a program development plan (PDP). The SPO SME explained that this effort was intended to provide the eventual contractor an in-depth understanding of the program requirements. The user SME recalled being consulted on the phone, and he formed the impression that the SPO and contractors were having difficulties nailing down the details of the requirements.

Contractor D was competitively selected to perform the development. They were teamed with two subcontractors. One subcontractor did software integration, while the other provided expertise in the joint system that received data from System D. The

contractor team proposed to leverage their understanding of the joint system to give system D a similar look, which would assist with compatibility and training. Contractor D started the program without experience in the Air Force legacy and host systems.

The SPO SME stated that the program was originally expected to be a simple development to replace the legacy system, implementing the same tools with new software and developing the interface to the joint system. Once the contractor started coding, it was clear that what the SPO referred to as the "business rules" for the system were not sufficiently understood. Business rules, referred to in this research as "operational steps", included the sequence of steps the user would take to operate the system, and how parts of the system should interact. According to the SPO SME, these rules captured the operational flows of the user, enabling the software designers to write the code accordingly. He felt that without definition of the operational steps, the system would not be operable by the user.

The lack of definition of operational steps by the government highlights the possibility that the distinction between requirements, operational steps and design details can become blurred during the requirements definition and design phases. The SPO strategy was to generate the PDP to define detailed requirements and to involve the user in a series of workshops and reviews to validate requirements and participate in design definition. However, the operational steps definition issue proved to be more problematic than had been anticipated during original program planning.

Capturing the operational steps took a large collaborative effort, involving detailed discussion and decomposition of tasks with the users. The stakeholders sometimes referred to the legacy system to see how things were being done in the field.

Adding to the design trade space, the contractor proposed more efficient operational steps for user consideration. At a series of workshops, Air Force agencies eventually reached consensus on the operational steps. The extra time and effort contributed to cost growth and schedule delay. The overall program stretched from 18 months to 36 months, of which 26 months constituted the design phase.

Requirements for each software segment were incorporated and validated over the course of three spirals. For each spiral, two working groups were held with the user, SPO and contractor. These sessions included the effort to wring out the operational steps. A third session, involving a final user review and testing of the software, completed the spiral. The next spiral incorporated additional requirements and repeated the cycle. This building block approach accommodated validation of requirements within the spirals, but it did not address system level issues such as interactions between software segments. The contractor depended on an internal system integration team to work these issues.

During the course of the development, Contractor D had an objective to define a single approach for the appearance and function of the graphical user interface (GUI) that would apply across the four segments. Users from all segments were vocal on how the screens should work, and a separate workshop series was established to provide definition. The User SME commented that these workshops were not managed well because some people who should have been involved were not there. This observation provides an indication that it was difficult to integrate the perspectives of the four user groups.

After the stakeholder interactive work sessions were complete, Contractor D spent several months integrating the segments together before presenting results to the

community at a final system review. Since this integration activity had design implications, it was treated as part of the design phase for purposes of this study.

The SPO and the users had three primary emphasis areas for Program D. First, they focused on defining and implementing operational steps. Second, the SPO shared an interest with the contractor in the development of a common Graphical User Interface (GUI) for the different segments. Lastly, all three stakeholders wanted to ensure the interface to the host system and the joint system would be functional and effective.

SR Description

Over the course of the collaborative sessions with the SPO and the user community, Contractor D employed increasingly sophisticated system representations (SR) to share developing knowledge about the system. The SR emphasis was on portraying screen appearance and the series of screens and steps an operator would go through to perform tasks. The primary benefits of the SR to the stakeholders were to facilitate development and capture of operational steps, to validate that the design met requirements, and to provide an opportunity for all stakeholders to give feedback and iterate on design details.

The first level of SR sophistication involved static screen shots and discussion of the sequence of screens that would be navigated by the user. This approach did not allow for any interaction with system software and had a limited capacity to convey the options and steps that would be associated with operating the system. The contractor used this approach in early sessions when software code was not yet available. The primary

benefit involved requirements clarification and first order thinking about how the user would operate the system.

Later meetings involved sharing a primitive version of segment software that displayed screens associated with requirements of a particular spiral. This software allowed for operator interaction and navigation between screens, but it had limitations. The different segments were developed separately, so the community was not able to assess how the segments would interact. Interfaces to the host software and the joint software were simulated. Also, it was not possible to get a sense of timing issues, such as how long the system would take to retrieve or process data.

The final level of system representation was an integrated version of the system software in which the segments interacted with each other. This SR was shared with the SPO and the user community during a final system review at the end of the design phase.

The contractor SME explained that schedule constraints forced a parallel segment development strategy. Developing a total system SR at the same pace as the segment developments would have entailed expending more resources and slowing down the program. The contractor depended on a small systems group to look at the system database and interfaces and track how the pieces were coming together.

The contractor SME didn't feel that the user community appreciated the system considerations since the users all focused on their separate areas. However, the user SME was frustrated that earlier versions of the SR had not given enough visibility into the overall design. User comments at the final system review could not be incorporated in time for this software release.

SR Usage

This section describes how the different versions of the SR were used during System D's collaborative sessions.

For each of the three spirals, the initial working group concentrated on requirements clarification. Participants were sent read-ahead materials, including lists of requirements and representations of screen shots, which were an initial form of SR. At the session, the contractor led participants through scenarios showing how the software would be used to meet specific requirements. The stakeholders found that the PDP was not sufficiently detailed, and they had to work through what the system needed to do in detail. This activity, which established the operational steps, was extremely time-consuming. The discussion was aided by viewgraph projections of the screen representations.

The second time the working group met, participants reviewed more requirements and additional screens. Development software associated with that spiral's requirements was used as the SR. The objective of the sessions was to continue building consensus on detailed requirements and operational steps. The contractor also welcomed SPO and user feedback on the GUI design.

Users were asked to sign off that the design was meeting requirements as it was presented in order to keep the overall development on schedule. The user SME was concerned with limitations in what the SR could show. It was not possible to see how data flowed between segments and out to the joint system or how the interface to the host system was being implemented. Also, the SR did not demonstrate how long the system

would take to do many tasks. Without this visibility, he felt that approving individual screen designs had limited value. He commented that better representation of the host and joint systems and a true beta software release would have been very helpful to more fully validate requirements and evaluate the design.

After the two working group sessions, a user review and test session was held for each spiral. These meetings were very structured. The contractor walked through each requirement and used development software to show how the screens would work and how screen sequences satisfied the requirement. The lack of visibility into interactions remained a concern for the user SME, although he felt the final product turned out to be "fairly decent" due to the skills and knowledge of the contractor team lead. The user SME also was concerned that the SPO did not seem to have a good system for capturing requirements that were deferred to future deliveries of the software. He believed that many such agreements were lost.

According to the contractor SME, stakeholder discussions resulted in many suggestions for better ways to accomplish tasks. Comments concerning the operational flow or how data was represented on the screen could usually be implemented easily and quickly, particularly if they were raised in early spirals. If a suggestion involved adding system capability, the contractor provided the impact to implement the change, and the SPO decided whether additional resources were warranted.

The contractor periodically held larger user reviews with all of the stakeholders, including senior government managers. They froze the system baseline and presented guidelines regarding what functionality was and was not present in the system. The contractor segment leads provided a day and a half of presentations on the screens and

functionality of the system. The last half-day was reserved for participants to sit at a workstation and have "free play" time on the system. These large reviews resulted in many comments, some of which were implemented immediately. Contractor D committed to respond to remaining issues and actions within one week. Limitations of the SR, as described above, still applied, although the community did get exposure to the work going on in other segments.

The final system review was held with the community after several months of contractor internal work in which the segments were integrated with each other. For the first time, the government saw segment interactions and was able to assess system timing issues. The user SME recalled that he had comments on design details that were not visible in earlier versions of the SR. He estimated that 20 – 30 percent of the overall system implementation involved aspects he would have liked to have seen done differently. He felt perhaps a quarter of these observations were critical. Unfortunately, users had to choose between delaying release of the software by several months to allow incorporation of comments or waiting for changes to be included in a future increment of the software. They chose to wait for the next delivery.

The stakeholders mentioned experiencing the following benefits of using the development software as a SR.

- Using the SR was critical to rapid resolution of issues. Once the contractor structured a design approach, stakeholders could see how an operator entered a state, what data was available, and what actions could be taken. Seeing the flow of the operational process enabled a faster consensus on whether the design was acceptable or what it would take to make it acceptable.

- The SR also helped when the program was responding to the challenge of falling behind schedule. It gave everyone a sense of the maturity of the system, and a way to quantify how much more work it would take to incorporate changes.
- Sometimes the contractor decided to make a case that they could or should accomplish tasks in a new way. The user SME indicated this proved to be difficult due to resistance in the user community. However, as users started to interact with the SR, they could appreciate the new capability that they were given or the reasons driving the change, such as increased compatibility with the joint system. Having the SR for these circumstances was described as essential in gaining user acceptance.

The contractor's approach to generating the SR was to prototype screens and menus quickly with software. Program D involved fixed menus, which made that approach easier. The contractor knew they were not able to create a seamless representation. They wanted to get closer and knew it would be useful, but in some cases technology or programmatic constraints didn't support creating a fuller system picture with the SR. All stakeholders agreed that having a partial picture of the system in interactive software form was far better than relying on the static approach of documentation and viewgraph representations.

Stakeholder Interactions

The primary SPO role in stakeholder collaboration was to build consensus between user elements and with the contractor for requirements and design issues. The SPO philosophy was to let the contractor lead the way with the design. The user SME indicated the SPO did not initially have anyone who deeply understood the intended functionality of the system. The SPO's sense of the importance and nature of the mission improved over time as they grew to understand user needs. All three stakeholders observed that inadequate staffing in the SPO was a limiting factor, sometimes delaying information exchange or decisions.

The user SME felt that there was a good rapport between the SPO and user, and the SPO was responsive on requirements issues. Management issues, such as setting up meetings or generating schedules or agendas, seemed to be more of a problem. One stakeholder felt the lack of management focus on orchestrating resources had some impact in delaying the program. There was no forum established for collaboratively working management issues, so integrating practices that might have been discussed or created were not considered. While a top level Integrated Product Team (IPT) was established, the user SME indicated it did not meet often enough, and no periodic telephone conferences were instituted to establish and disposition management issues. He observed that the IPT only met when there was a crisis.

The SPO insisted that Contractor D document operational steps for the system, while the contractor's original expectation had been that the government would do so. The contractor took on this role, but found that having four segments, each with their

own user communities, made the set of operational steps that emerged somewhat fragmented. It was difficult for Contractor D to drive a vision of the integrated system given diverse user perspectives. A stronger government systems engineering role, including close collaboration with the SPO and all of the user elements, might have helped the contractor create a more coherent, integrated system.

The user's primary role during development was to define an official set of requirements for the program. This authority included decisions to move elements to a later spiral. Since user representatives knew the old system and could offer perspective on how they anticipated using the new product, they were able to assist in defining operational steps.

User representatives tended to have two types of observations, based on their experience with the legacy system. In some cases, users would describe the way things were being done at present and insist the new system should be implemented in the same way. In other cases, users knew of aspects of the legacy system that were undesirable, and they wanted improvement.

When the contractor proposed an innovative approach, users tended to resist change. However, sometimes changes were necessary in the interest of defining system-wide standards or ensuring joint requirements could be met. In other cases, the contractor may have discovered ways to give more flexibility or capability to the operator. All three stakeholders agreed that having an interactive SR seemed to be very powerful in helping users evaluate and accept proposed changes.

The government stakeholders saw the contractor as being very motivated and responsive. The user SME felt they really took the user's interests to heart. They were

interested in forging good relationships and teams, and they wanted to make progress. The government was not typically held up by anything that was within contractor control. Contractor D reacted to many changes without needing formal letters, and the community was willing to catch up later on formal changes to the PDP in the interest of keeping progress going.

The contractor to user interaction was described as comfortable, although it could become problematic. The SPO acted as a referee between the parties when issues arose. Direct interaction of personnel at the working groups was encouraged, and after people got to know each other they called with questions in between meetings. The parties were willing to operate through informal channels rather than passing all communications through the SPO, which sped up the access to information and helped the contractor define and deliver what the warfighter needed.

The contractor SME indicated they tried not to let strict interpretations of the contract interfere with understanding the user's real needs. In cases where new requirements emerged, the parties would reprioritize and trade off requirements as necessary to stay within budget constraints. Sometimes additional funding was secured. The SPO recognized and accepted that Program D would be an evolving program.

System level requirements changes were coordinated between the SPO and users. The contractor was not directly involved. They were sometimes consulted on the implications of requirements changes, but this was not done consistently.

Adaptability

The number and value category of Case D's collaborative changes are provided in Table 5.4. Value categories of collaborative changes are explained in Table 4.6.

Number of collaborative changes	9
High value changes	5
Medium value changes	4
Low value changes	0

Table 5.4. Collaborative changes for case D

The collaborative changes fell into the following categories: three involved technical performance; three concerned the user interface; one addressed interoperability; and two related to reliability.

The collaborative changes for this program included the following:

- Centralized a data function to avoid accidental corruption of data by multiple users
- Increased user flexibility to view assets before and after deployment
- Applied global update or conversion of data files to analysis files to ensure consistency of data
- Enhanced user capability to save and retrieve past data
- Created an interface with another data system to ensure correct data transfer
- Added capability to do mass changes to data

The focus on only one of four segments in this case raises the question of whether it is comparable to the other cases, or whether the number of collaborative changes may be an understatement of the program's actual degree of adaptation. The analysis and reporting segment represented a microcosm of the program in which one user representative integrated requirements of a particular user community and interacted with the SPO and the contractor. In this sense, Case D was a comparable scenario to the other cases. However, looking at one segment may have resulted in missing some collaborative changes that impacted either the entire system or interactions between other segments and the analysis and reporting segment. Therefore, it is possible that the observed number of collaborative changes is not as complete a capture of program adaptations as was carried out for the other cases.

Within the context of the eight case studies performed for this research, program D was considered moderately adaptable, falling in the third tier of adaptability. As described in section 6.2, four of the cases exhibited more adaptability, one exhibited less adaptability and two other cases were also rated as moderately adaptable.

Summary

Program D put time and resources into creating and using partial software builds as system representations in a series of collaborative sessions held throughout the design phase. Having four segments with different user communities made knowledge transfer and assimilation difficult for all parties. The stakeholders indicated that usage of software SRs was very beneficial to the program when compared to traditional means of sharing information. However, the SR lacked some essential elements that made

complete evaluation of the design difficult. Contractor D did not generate a system-wide system representation until the end of the design phase, when it was too late to incorporate comments.

The program achieved a relatively small number of collaborative changes compared to other cases, although more than half of the changes were considered to be of high value. The substantive effort that was required to capture operational steps took up a great deal of program resources and put cost and schedule pressure on the program, reducing opportunities for adaptation.

Lessons learned for Program D included the following:

- Programs that replace legacy systems have to deal with user expectations on how the new system will be operated. These programs should either involve users in definition and capture of operational steps before contract award, or budget for this effort to take place during requirements clarification and early design.
- Validation of requirements is difficult in spiral development in the absence of insight into such aspects of the design as timing, segment interaction and interoperability with other systems. Program planners need to determine means of simulating external interfaces and estimating how long the system will take to perform certain functions. Alternatively, they may need to adopt a method of validating requirements that allows reviewers to assess these aspects of the design and provide feedback as full insight becomes possible.
- The partial software SR was useful in evaluation of operational flow of the system, rapid resolution of segment-level issues, evaluation of the maturity of the system, and user acceptance of innovative new approaches to doing tasks.

- Inadequate SPO staffing levels contributed to delays in information exchange and decisions.
- Programs with multiple stakeholders need a forum to discuss management practices and decisions.
- The contractor was at a disadvantage in the role of integrating the perspectives and requirements of multiple users. A government agency that can be free of conflict of interest and can exercise a sufficient span of control should play a systems engineering role when a program has requirements from multiple user communities.
- Informal communication, willingness to act now and document later, and the ability to reprioritize requirements sped up information sharing and facilitated adaptation.
- One of the stakeholders (typically the SPO) needs to keep track of agreements concerning program changes that will be implemented in future increments.

Case D had the advantages of strong user involvement and good collaborative practices between stakeholders. However, the program faced financial pressures due to the added work associated with defining operational steps. Program management concerns included the lack of a forum for management issues, understaffing in the SPO and dependence on the contractor to lead integration of the operational steps of four disparate user communities. Another problematic factor was that the SR did not provide a representation of the full system until the end of the design phase. These financial and

management considerations, along with limitations in the SR, may have constrained the level of adaptability of the program, which was moderate compared to other cases.

5.6 Case E

Case E focused on a small, six-month design effort for a new release (referred to here as “Version 3”) of a large software program. The program provided command and control functions for Air Force assets operating in a theater. During the timeframe that Version 3 was in design, one version of the program (“Version 1”) was already fielded and another (“Version 2”) was in testing.

Version 3 was intended to be a rapid effort combining fixes to problem reports from previous testing and less than a dozen low risk initiatives related to software applications. Both the program’s schedule and its fixed budget limited the content the prime contractor (Contractor E) could deliver.

Per the SPO subject matter expert (SME), the stakeholders revisited Version 3’s requirements as the design effort progressed. When the program lost some of its funding, the requirements had to be adjusted to fit the new budget. Problem reports from testing of Version 2 sometimes were inserted into Version 3’s scope of work. The SPO SME indicated that stakeholders got involved in “horse trading” which involved adding new, higher interest requirements in place of others. One of the fundamental tensions in this process was that the SPO emphasized resolving problem reports to make the software more usable while the user headquarters (User E) was more interested in added functionality through new or improved applications.

Three categories of users had important involvement in the design phase. User E functioned as a headquarters office, integrating requirements for the user community. The second group was a cadre of operators involved in testing of Version 2, who had

interactions with the SPO and the contractor. The third category of users included field operators whom the SPO consulted on a mostly informal basis. These three elements of the user community played different roles in Version 3 development, as described below.

User E was very engaged early in defining requirements for Version 3. Once the contractor started design work, the SPO observed that User E had minimal interaction, consisting primarily of discussing requirements or funding issues or getting schedule status. The User SME felt there were two factors that diminished involvement by his office. The effort was low priority compared to Version 2 and other efforts that User E was tracking. Also, he observed that some elements of SPO management resisted interaction due to the belief that the user role was limited to providing requirements.

Both the SPO and contractor E took advantage of having operators available who were testing Version 2 during the Version 3 design phase. The SPO SME noted that he developed a sense of what the user wanted from the system through frequent discussions with the testers. The contractor SME related that testers were sometimes involved in evaluation sessions to look at aspects of Version 3 and give feedback.

The SPO SME also involved field unit personnel in evaluation of some issues. The SPO retained a contractor with system administration expertise at one operational location, and frequently involved this individual in technical discussions with the contractor. User E personnel did not have past experience in operating the system, so the SPO project officers and the contractor created informal channels to users whose perspectives could be of value to the new release. The SPO SME called field personnel to get guidance much more rapidly than User E could get a formal position. In general, though, field users were deployed and were so busy that it was hard to get feedback from

them. The SPO SME indicated they were trying to find a balance in which SPO programmatic concerns, contractor advice and the user perspective could all be weighed when making decisions.

The SPO assigned two project officers who could give technical feedback to the contractor. They had user backgrounds in two different functional specialties related to the program. Both project officers recalled visiting the contractor's plant about four times to interact with the Version 3 development software (the system representation, or SR, for the system). The SPO SME noted that User E personnel did not travel to the contractor plant for sessions related to the Version 3 design. He felt their lack of operational experience would have made it difficult for them to assess the design.

Both the user and SPO SME's had a positive impression of Contractor E's involvement and capabilities. According to the user SME, the contractor was very active in seeking to understand user needs, talking with User E personnel constantly on system level issues. He noted they had competitors for future business, which may have provided additional motivation. The SPO SME indicated that the contractor was very knowledgeable in several technical areas including the Air Force's command and control infrastructure, associated databases, and joint requirements.

The contractor SME described that typical design efforts included intensive review sessions with the SPO and user for each new version of the software, both to lock down the requirements and to meet several additional gates in the design and testing phases. The review process was highly formalized and included substantive documentation of the software design. Given the shortened timescale and relatively low complexity of Version 3, Contractor E used a scaled-down approach, involving the user

and SPO primarily in sessions for individual applications. User involvement took the form of interaction with Version 2 testers who were available at the contractor's plant. The contractor SME indicated there were not really any new aspects of interaction between applications that would have justified system-level scrutiny with the user.

The user SME stated that the first time he saw development software was at government in-plant testing. Before that, his interaction with the program had been through telephone conferences and technical meetings. However, the in-plant testing was one month before the start of developmental testing, which was too late to make any significant changes. The user SME felt that the design could have benefited from user feedback if he could have seen and interacted with the code as design decisions were being made along the way. He indicated this type of interaction was happening on larger projects and in other forums that brought the user, SPO and contractors together.

The contractor SME observed that having the system in the field was helpful for future developments. He indicated that having a good foundation of operational experience in the field helped the contractor determine what information the system needed to display to support command and control decisions.

For this program, the SPO emphasis was on the user interface and system timing (e.g. how long would it take to update a large data set.) Interoperability of the system with Version 2, with other Air Force systems and with other services was also critical. The SPO SME observed that User E seemed to put a major emphasis on keeping to the defined, rapid timeline for delivery. He felt Version 3 was a proof of concept for User E that such a quick incremental delivery was possible.

The user SME stated that the top priorities for this effort were the user interface and reducing the burden on systems administration. He described Version 3 as a cleanup build. The user was also interested in web enabling of applications, and the user SME indicated that the stakeholders had some good discussions at the working level to reach common ground on how to approach this new technology. He felt it would have been harder on a larger, higher visibility program to have these types of discussions. The SPO SME agreed that many valuable offline discussions had taken place on web enabling. From his perspective, the SPO provided User E with a better appreciation of how much it took to web enable an application by explaining the underlying technology.

SR Description

The development software was used as a SR during design of Version 3. Both Version 2 testers and the two SPO project officers had opportunities to interact with the SR at the contractor's plant. User interactions focused on applications that were being reviewed as part of the Version 3 development. The SPO interactions took place on an informal basis.

The SPO SME remembered looking at a first prototype of the software about two months into the design effort. He commented that it was not really ready for operator interaction, and he didn't get much out of the first session. He recalled that the contractor was nervous about sharing the product so early, but the SPO understood the contractor's explanation of what capabilities and fixes were in the system at that time. The contractor SME recalled that the four SPO sessions had probably been with different

builds of Version 3, each with specific content and goals, so the SPO was probably exposed to different applications during different sessions, rather than seeing the full set of applications four times.

SR Usage

SPO and user interactions with the SR for Version 3 were very limited, and happened on an informal basis. The short (six month) design window, low priority, and perceived low risk of the effort made this version different from other version design phases in which the stakeholders indicated more in-depth and formalized interactions with the software took place.

The contractor SME recalled that user interaction was on an application-specific basis when Version 2 testers were involved informally in ongoing contractor development sessions. These sessions included interaction with the SR, and the contractor SME recalled that the user made significant inputs on two applications that they reviewed. Some user comments resulted in formal change requests that were put into the system. If these changes had sufficient priority, they were considered for inclusion in Version 3 or a future software build. The contractor SME stated that these sessions might not have shown up on a formal schedule, and the SPO may not have known about some of them. He felt it was a contractor leadership issue to determine how much initiative they should take to seek out user input in this fashion.

When the SPO SME traveled to the contractor plant, he made it a habit to devote a day to interacting with the system software. He sat down at a workstation with contractor test personnel and the program manager. In addition to running through draft test

procedures, he tried things on the system on an informal basis. In one example, he described pushing information between himself, playing one functional role, and the other SPO project officer simulating another role, to see how the system would react. The SPO SME recalled that these sessions resulted in documented comments that, in general, impacted the look and feel of the program, improving operability for the user. Examples of this type of input included reducing the number of actions the operator would have to make to accomplish a task or ensuring the system would provide a status screen if it was in the middle of a calculation so the user would know what was happening.

The gist of a typical discussion with the contractor at these sessions would involve what state they got into with the system, what assumptions the designers were making, which assumptions were wrong, what should be changed, and how they could work through the problem. The SPO SME recalled finding problems by going through these types of discussions, which would be formally documented for disposition. He stated that these sessions were the only chance the SPO had to look at the system during the design phase. By contrast, he could look at briefing charts that implied the system would work, but conveyed no real understanding of the design implementation.

The SPO SME focused his review of the design on two factors: what the interface looked like to the user, and how well the code performed from a time sense, i.e. how long it took to do things like updating a large set of data. When performance issues were identified, the question was what to trade off to improve performance. Both the SPO and the contractor were conscious of budget constraints when thinking about design options.

Stakeholder Interactions

Interviews with the SPO and user SME's revealed that the two organizations had a distrust and low regard for each other during the course of the Version 3 effort. For the most part, they were not working closely together and did not share knowledge of each other's needs and constraints. The following examples taken from interview discussions about similar themes illustrate the different viewpoints between the two organizations.

- On communication and collaboration:
    - User SME: the SPO shifted from a collaborative approach that he had observed earlier in the program to limiting interaction after getting user requirements. The user SME preferred constant communication to avoid surprises during testing, and he noted that user priorities were constantly in flux due to real world changes. The SPO was taking a week to get back with him on decisions that he felt should have been resolved within a day.
    - SPO SME - User E wouldn't work with the SPO to solve problems. They approached one issue by contracting with a provider directly, not trusting what the SPO was saying should be done. The user never communicated the real root issue to the SPO, who had to figure it out by inference.
    - Observation: both parties distrusted the other to the point of avoiding collaboration that could have been mutually beneficial.
- On implementing an off-the-shelf application that the user had discovered:
    - User SME: the SPO said we were specifying a solution instead of stating requirements. We responded that the application was free and it existed now.

The SPO claimed it was poorly engineered, and they could get the contractor to provide the same thing in 18 months.

  - SPO SME: the user wanted a new application and specified an existing solution. The user didn't understand the integration implications for Contractor E. When the user intervened at the flag officer level and forced the program to integrate the application, it didn't work right, and costs escalated.
  - Observation: the user did not understand SPO and contractor constraints, and the SPO did not adequately communicate the reasons for their position. The user SME commented at one point that if the SPO explained their constraints and had good rationale, the user would accept their position. This level of communication was not taking place between the two organizations.
- On participation of User E at meetings:
  - SPO SME: User E had other priorities and often backed out of meetings or telephone conferences where they were needed to make decisions.
  - User SME: Missing meetings got the SPO upset. It wasn't clear why the user needed to be there and what decisions they needed to make. If they were reviewing a new diode, the user doesn't care.
  - Observation: the SPO hadn't made the case that user E involvement was important so User E chose to put time into other priorities.

Other evidence of a rift between the SPO and User E came up in the interviews. The user SME felt that the SPO was more concerned with the pursuit of engineering excellence than they were with the user's needs. He observed that things that were not

critical to the user become paramount issues for the SPO because of their focus on engineering. Since resources were limited, the User SME emphasized that user priorities had to come before SPO preferences. He felt the contractor understood better than the SPO what the user wanted.

The user SME wanted the SPO to communicate a vision for the program and challenge the contractor to determine what was possible rather than telling the contractor how much money they had and asking what they could get. User E felt he might have been able to justify a request for more funding if the contractor came up with innovative approaches to meet important objectives.

The contractor SME had the impression that SPO technical personnel could sometimes be impractical and miss real-world aspects of program decisions. He felt the SPO lacked initiative to collaborate with the user, perhaps because they were attached to the current plan for the program and it was too hard to change.

From the SPO perspective, one reason for the minimal interaction between the SPO and User E was the lack of operational experience in the User E office. When the SPO prioritized open problem reports, they would depend on discussions with Version 2 testers and SPO technical experts. The SPO SME indicated that the User E contact would typically rely on SPO expertise and endorse the priority list without comment. However, User E took a stronger interest and role in approving or delaying insertion of technology for new applications.

The SPO SME's connections with the field allowed him to get four or five opinions in an hour over the phone. He noted that user E's mode of operation involved sending messages to Air Force major command headquarters, which would take

“forever” to get answers. User E wanted answer in writing while the SPO SME wanted guidance, and was able to get it through informal channels. The SPO SME wanted to understand how operators in the field were doing the job, rather than relying strictly on user requirements.

SPO project officers also maintained connections with other government program managers to understand interoperability issues with other systems. They would then work any issues with the contractor.

The SPO SME related that the number of personal connections he had became a problem. He got 300 – 400 emails a day, and his voicemail filled up. He described the situation as communications overload. He also indicated that the work pace on the program caused “burn out” for him and others.

The contractor SME felt working relationships with the SPO were very good. He found meetings with the SPO to be extremely effective for getting things done. The SPO SME also felt that the SPO had good synchronization with Contractor E, which had a large, positive effect on keeping the program running. The contractor was willing in many cases to make changes and accept a follow-up contracts letter. This informality made for faster decisions and implementation of changes.

Several other contractor roles were highlighted in the interviews. The contractor had the primary expertise to establish what was technically possible, how long it would take, and how much it would cost. The contractor SME described that one of the contractor personnel had maintained an informal connection with a contact on the joint staff at the Pentagon, which provided a source of information on joint requirements. Contractor E also helped the government write Concept of Employment documents

(COEs), and held formal reviews with users for validation. The COEs described individual actions of an operator in an operational context, the number of operators, how they used applications, who had what screens open, etc. The contractor SME indicated this information provided boundaries and goals for the design. One example was high central processing unit (CPU) usage, which could only be managed by understanding in detail how the system would be used. The contractor SME described the COEs as in part a defensive maneuver to provide written boundaries so the testing environment would be realistic.

The contractor relationship with User E was fairly strong. The contractor SME stated that his corporate guidance was to understand exactly what the user wanted, and as a result they were very involved in planning for the future of the program. The SPO reaction to this interaction tended to be personality dependent, with some individuals having huge discomfort while others had no problem. The contractor SME emphasized that when a user spoke with the contractor, any formal direction still had to come through the contract and the SPO.

Integrated Product Teams (IPT) for program E were structured with counterparts from the SPO, the user, and the contractor. Per the contractor SME, a key to success was having one contractor expert (retired military) with past user experience on each IPT. The experts spent time with developers showing them how the software would be used.

The contractor had a formalized process that identified points when the IPT held reviews. Materials were provided to all stakeholders and phone dial-in was made available. The contractor made sure the user was involved in requirements reviews, but during design he observed these reviews were “the furthest thing from the user’s mind.”

The SPO was involved in reviews during design. The contractor SME felt that User E tried hard to arrange for field people to participate in some of the reviews, but the main challenge was to maintain consistency. Different user representatives had varying perspectives, which was sometimes difficult for the program.

Adaptability

Case E's collaborative changes are broken out by value category in Table 5.5. Value categories are explained in Table 4.6.

Number of collaborative changes	12
High value changes	2
Medium value changes	10
Low value changes	0

Table 5.5. Collaborative changes for case E

The collaborative changes fell into the following categories: two enhanced technical performance, seven concerned the user interface, and three involved interoperability.

The following are a subset of Program E's collaborative changes:

- Changed algorithms for building a product so the operator could do so faster and with less effort.
- Changed a data item within the application and database to accept positive or negative values.

- Modified an application to accept data from another military system.
- Simplified the algorithm for one application, speeding up response time.
- Added capability to import and export a new file format.
- Provided mechanism for subordinate units to provide more accurate capability data to System E.

Compared to the programs studies for this research, program D was considered moderately adaptable, falling in the third tier of adaptability. As described in section 6.2, four of the cases exhibited more adaptability, one exhibited less adaptability and two other cases were also rated as moderately adaptable.

Summary

This case uncovered a range of interactions between stakeholders. The formal interaction between the SPO and User E was minimal, although the SPO compensated for this situation through informal interactions with field users. User E indicated that he was discouraged by the SPO from talking with the contractor, but he still had informal contact, particularly to clarify system level considerations.

The pattern that emerged was that stakeholders who lacked formal pathways of interaction managed to forge informal mechanisms to obtain insight into the perspectives and knowledge of other stakeholders. The disadvantage of this situation, as shown in this case, was that stakeholders who were not communicating tended to mistrust and denigrate the actions of the other party, in part due to a lack of understanding of their

needs and constraints. Better collaboration between the SPO and User C would have allowed them to pool their knowledge to the benefit of the program.

The following lessons learned were observed for Case E:

- The SPO needs to make the case for user involvement in design so that the user will be more motivated to participate and stay connected. Maintaining close SPO – user interaction during design allows identification and evaluation of potential adaptations and lets the parties keep track of each other's shifting needs and constraints.
- Informal lines of communication can provide valuable knowledge sharing opportunities. However, formal interactions should not be neglected. The SPO and user E lacked understanding of each other's needs and constraints, which seemed to damage trust and respect between the parties.
- Testers are one source of operational experience that can be of benefit to designers in understanding how a system will be used. Sometimes testers may be more readily available than deployed operational users.
- The SPO needs to ensure the opinions of its technical experts are not overriding user priorities regarding technologies that are being developed or included in a program.
- Contractor hiring of retired military personnel with operational backgrounds is one mechanism for injecting operational perspective into a development program.

Due in large part to the relatively small scope and low priority of the program, User E did not take advantage of the opportunity to assess and influence the design even though a SR in the form of the development software was available to provide design insight. The SPO had minimal interaction with the SR. Several factors, as described

above, led to a strained relationship between the SPO and User E. Both the negligible usage of the SR by the government and the lack of substantive collaboration between the SPO and User E may have contributed to the program's moderate level of adaptability compared to other programs.

5.7 Case F

Case F involved the design of a software increment that provided a new framework and some basic functionality in support of the employment of a wide range of weapon systems. The framework, which provided a virtual operating environment to move into and between applications, was intended to support future software increments with additional operational capabilities. The program's initial beta software releases focused on the framework. This case focused on the design work associated with two later beta versions, which added a basic level of operator functionality. The overall program development budget was $53 million, and the design effort considered by the case encompassed a 21-month period.

SPOs from two different services provided government oversight for Program F. Only the Air Force SPO was interviewed for this case study. The two SPOs worked together and interacted with the prime contractor, third party software providers, and various user organizations through an Integrated Product Team (IPT) structure. The Air Force SPO worked with an internal integration office to connect with weapon system programs and explore cross cutting issues relevant to Program F. SPO personnel went to exercises and talked to operators of the legacy system (the precursor of Program F) to get their perspective. The Air Force SPO also drew on knowledge of the legacy system through a co-located contractor who had participated in its development.

User requirements were integrated by an Air Force major command headquarters office (User F), which collected inputs from headquarters counterparts representing the affected weapon systems. User F maintained contacts (for example an operational wing representative) to clarify requirements and priorities. The user subject matter expert

(SME), who worked for User F, indicated that a typical operator focuses on what he needs to do with the system. He felt user representatives from the field who were exposed to the Program F design typically evaluated aspects of the Graphic User Interface (GUI) related to accomplishing tasks. They were not interested in system design issues such as the way different applications moved data around, or how the data were collected and integrated. User F, as the headquarters representative for all user organizations, had a broader perspective that included system level considerations such as how the applications could best interact with each other.

Contractor F was responsible for developing the new framework and some functional software as well as integrating and testing functional software developed by other contractors. Subcontractors supplied some of the functional software, but a large portion was provided as government furnished information (GFI), i.e. from separate government contracts. One important ingredient in the program was direct and regular communication between all of the software developers.

The program lacked an approved Concept of Operations document from the user, so Contractor F developed “use cases” and scenarios to define what the user would do when employing the system. The two SPOs and various user representatives reviewed these constructs to make sure they made sense. The contractor used this insight to help design the software.

Program F’s heritage included two development programs with related capabilities. One, a currently operational Air Force system, or legacy system, was very comprehensive in meeting user requirements, but the user community objected that it was hard to use and maintain, and was prone to crashing. Program F was intended to provide

the foundation for a replacement system. The other related program was developed on an ad hoc basis by the Air Force, in the sense that requirements were generated from periodic user conferences rather than through formal requirements documents. It put an emphasis on ease of use and rapid addition of new user requirements, which made it very popular in the operational community. However, due to its architecture, it had growth limitations that could only be overcome with a new development. The Program F architecture was intended to make future upgrades easy.

The ad hoc program was used as a design baseline for Program F. One stakeholder explained that the vision for Program F was to create the same advanced capabilities as the legacy system, but with the intuitive look and feel of the ad hoc system. The SPOs and the user community instructed the contractor to emulate the ad hoc design when making design choices. The contractor had to make design decisions primarily when the ad hoc system did not provide a design template, although the user SME commented that System F included increased flexibility for the user in such areas as sizing windows, moving objects around on the screen and defining new button functions.

During the first half of the design effort, management decision-making was slowed by diverse priorities between the two acquiring services and a bureaucratic IPT structure. One service was driven by schedule considerations. The other service had no time pressure, but considered cancellation of the program when cost and performance became concerns. The contractor and SPO SME's both felt that even simple decisions were slow and painful, as the two SPOs and various IPT elements had to be aligned. The possibility of program cancellation created tension between the services and with the contractor.

Several factors contributed to cost and schedule difficulties in the program. The stakeholders felt that the job had been underestimated from the start. The contractor was receiving input from both SPOs and from diverse parts of the user community. Per the contractor SME, a lot of churning took place in the first half of the program.

The Air Force SPO initiated a series of program audits, which identified problems on the program. As a result of these observations and some other factors, the cost, schedule and technical baselines were restructured halfway through the case study period. Also, management practices were improved. The number of IPT's was reduced from 12 to 4 and the process for reaching technical and programmatic decisions was streamlined. In the same timeframe, concerns regarding program cancellation were alleviated, increasing trust between the program participants.

Contractor F continued to face financial and technical pressures after the re-baseline. One SME commented that the contractor seemed to react to this pressure by becoming less responsive and more adversarial. The new structure included a liability ceiling, beyond which the contractor would be responsible for all expenditures. Several requirements that were not in the original specification had been identified at a technical meeting that assessed equivalency between the contractor's design and the ad hoc system. The contractor had identified a set of cost reduction initiatives, which were evaluated and approved by the IPT structure. Combining the new requirements with the identified savings resulted in a no cost contract change. However, the contractor SME observed that after the new technical baseline was established, there was no room to incorporate further technical changes within the cost constraints of the program, since there were no more expendable requirements that could be traded away.

SR Description

The system representation (SR) for this program was the development software, as it existed at various points during the design. The SR took on three different forms. During the latter half of the design phase, the development software was distributed on a weekly basis. At certain special events such as user's conferences, software demonstrations were conducted in which users could interact with the software and provide feedback. Also, formal beta releases of the software were made at prescheduled intervals. The SR included prime contractor software and third party software as it became available.

The weekly form of the SR was constantly being added to and subtracted from. This enabled interested stakeholders to determine what the system was not doing as well as what it was doing. As the contractor made design choices, the visibility provided by the SR enabled stakeholders to provide feedback in a timely fashion.

Government testers and SPO personnel routinely received the weekly builds and interacted with them. The third party software developers also benefited, frequently identifying interface issues and participating in their resolution. The user SME indicated that User F did not have the staffing to assess these builds.

Two aspects of the program structure delayed visibility into design aspects, diminishing the usefulness of the SR to the user community. Since the framework development (how data would be handled and presented) was worked up front, most of the system functionality, which was the primary interest of users, was not visible until late in the program. Also, system integration aspects of the design were not visible until close to the end of the design phase. The user SME was concerned that they might miss

several design issues related to how applications could interact, since they would not see an integrated product until they got into testing. He commented that there was no way to know how significant the integration choices were going to be.

No arrangements were made to give any field user representatives access to the software, except during planned demonstrations. The user SME commented that they hadn't had a product that they would want an end user to view. The concern was that user feedback would have been negative given the lack of expected functionality, which might have killed the program.

The contractor SME felt that the SR "made all the difference in the world." Without the SR, the contractor felt they would have had a "huge surprise" by the end of the program when the contractor design was mismatched with government expectations. Unfortunately, the case study took place before results of testing were available, so it was not clear how well the design was going to line up with user expectations. The contractor also stated that the SR provided evidence of progress with the system design, which was essential to prevent program cancellation.

SR Usage

The SME's described different patterns of SR usage for the demonstrations, the weekly builds and the beta software releases. Demonstrations were done at two major users conferences and at a design review. Weekly software releases were made available beginning about halfway through the design phase. Two beta software versions were designed and one was released during the timeframe covered by this study.

The first demonstration was done at a major user's conference with a software version that included the full framework, but no real functionality. Field users were brought in and asked to interact with the SR for about 30 minutes each. Survey forms were collected from the users. The SPO SME commented that if it looked like the existing ad hoc system, the users tended to be happy. The users seemed engaged by the opportunity for hands-on interaction, and they were interested in providing feedback. The contractor SME felt that positive feedback from the user demonstration was significant in saving the program from termination. He also indicated that the user feedback was valuable and influenced the design. The user SME recalled the demonstration as mostly a presentation in which the user could see the GUI but could not see what the system was intended to do since most of the buttons didn't work yet.

At the program design review, the software version that was demonstrated included some basic functionality. User feedback indicated the system design was overkill in some areas and should be scaled back. According to the SPO SME, this demonstration was the only real user interaction with the SR during the first of the two beta version developments encompassed by the case study. The next beta version was also taken to a user's conference as it neared completion. As with the earlier conference, users were allowed to interact with the system and give feedback through survey forms.

The weekly software builds were evaluated by both SPOs and by some members of the test community. Of these groups, only the Air Force SPO was interviewed as part of this case study. The SPO indicated they were tracking the number of interface changes in the framework, the accuracy of data, the quality of coding standards, and consistency between design documentation and the actual code. They also put emphasis on graceful

degradation of the design, for example ensuring data would not be lost from an errant keystroke. Most of this interaction involved anomaly resolution or design and documentation practices rather than identification of potential collaborative changes. The Air Force SPO also made use of the updated code to debug their internal code, aiding integration between SPO and contractor software.

The first beta release covered under the case study was evaluated by the two SPOs and the test community, but not by User F (the user headquarters office) or field users. The testers generated deficiency reports that were dispositioned by the SPO and contractor. According to the SPO SME, design work on the second beta release that was studied was complete several months before the planned release, which would make incorporation of post-release comments more problematic.

The delay of functionality in the SR caused User F to adopt a strategy of waiting on the test phase to wring out user-related issues rather than seeking user interaction with the SR during design. Both the SPO and User F felt that showing field representatives a partial product without much of the functionality expected for the final version would have put the program, which was already receiving scrutiny for cost problems and schedule delays, in jeopardy of cancellation. The user SME felt backed into a corner given the loss of time to identify and resolve potential issues with the design.

Stakeholder Interactions

The Air Force SPO's management philosophy was hands-off until the program started to encounter cost and schedule difficulties. After external audits of the program,

changes to management practices (including streamlining the IPT decision-making process) and the program re-baseline, all of the stakeholders became more aware of cost and schedule considerations, and program performance improved.

A senior manager in the Air Force SPO used his contacts to stay in touch with users' perspectives and to refine program requirements accordingly. This individual had been the originator and guiding influence over the ad hoc system that became the design baseline for the program. Interviewees felt that the senior manager's leadership and insight into user perspectives was of great value to the program. One SME commented that he was a force in driving consensus, but he also made decisions in the absence of consensus rather than allowing endless analysis. The user SME indicated that the SPO sometimes followed end user input and changed program requirements without coordinating with User F. This practice caused some tension, but was not considered a major concern.

The Air Force SPO participated in weekly teleconferences with Contractor F and the third party software developers to discuss design and interoperability issues. One example was discussion of a collaboration mode of the software in which multiple users could work a task together. In the teleconference, the parties discussed who could make changes and how the final product of a task would be locked in place. Design details like these were frequently hammered out in this forum.

Prior to the program re-baseline, too many people had been involved in the IPT decision process, often working at cross-purposes or across IPT boundaries. Management changes included definition of responsibilities and streamlining of the IPT decision process. When issues arose, the technical or management IPT's stepped in

quickly. The technical IPT met weekly to go over issues, getting answers within 5 working days. Contract issues were passed to the management IPT for resolution. The contractor SME emphasized that timelines of decision-making and ability to make decisions stand were critical to development and modification of the SR. The ability to predict timing of decisions gave him more control and was even more important than speedy answers.

User representatives focused on requirements of the program, rather than getting involved in design issues or programmatic factors such as determining the number and content of beta releases. Users from both services sat on a requirements review board for the program, which met quarterly to disposition proposed specification changes. The User SME noted that User F was not as involved in the IPT's as he would have liked due to other responsibilities. User F's priorities were influenced by the expectation that the contractor would use comparison with the ad hoc system to resolve design issues, mitigating the need for frequent user involvement. Contractor F observed that both services depended on testers to inject user considerations into the program. Given the scope of this study, it was not possible to assess the effectiveness of this strategy.

One major concern voiced by the user SME was his impression that the contractor didn't understand how the user would use the system and what aspects of the design might be good or bad from the user perspective. With the exception of large user conferences, the contractor did not interact with the true end user. The user SME felt the contractor was more focused on achieving the "look and feel" of the ad hoc system versus paying attention to how the user could get the job done without frustration. The SPO heightened the contractor's awareness on these issues after receiving user feedback.

A performance working group was started to compare System F with the ad hoc system as the design matured to make sure efficiency and speed of applications was considered.

The SPO SME emphasized that the user was primarily focused on near-term needs rather than planning considerations such as future system interfaces or technology insertion. He indicated that convincing the user to address obsolescence issues (like upgrading the software operating system) was difficult unless a crisis was imminent.

Adaptability

Program F generated 10 collaborative changes during the timeframe associated with the case study. As shown in Table 5.6, only two changes were “high value” (as defined in xxx), three were moderate value and five were low value. Five of the changes were in the category of user interface changes, while two were to reduce development cost, two concerned interoperability, and one enhanced technical performance.

Number of collaborative changes	10
High value changes	2
Medium value changes	3
Low value changes	5

Table 5.6. Collaborative changes for case F

Here are some of the collaborative changes noted for this program:

- Identified a tool developed by one of the third party suppliers to use in lieu of a tool developed by the prime contractor.
- Added ability to select multiple objects on the screen and drag them to another location.
- Determined the set of printouts that could be generated by the system.
- Provided synchronization to external databases – met requirement with minimum added capability (this was a development cost-driven collaborative change.)
- Provided ability for operators on different stations to see some portions of each other's work. Determined minimum acceptable capability (again a development cost-driven collaborative change.)
- Provided a means to run the system with a subset of full capabilities on a non-certified platform.
- Added capability to import/export data in the format used by another, related software system.

Of the 8 programs studied for this research, program F was the only one considered to have low adaptability. As described in section 6.2, all seven of the other cases exhibited more adaptability.

Summary

Adaptation was difficult on this program for a number of reasons.

- Overly bureaucratic IPT processes slowed early government decision-making, and inputs from multiple stakeholders caused some churning in the contractor's efforts to make progress during the first half of the program.
- Schedule delays and cost growth put pressure on the contractor to perform to the minimum specified technical baseline.
- By the time management reforms were put in place, the new cost and technical baseline had been established and, the contractor SME felt the remaining requirements were all high priority, making it difficult to consider requirements for deletion in favor of any new ideas.
- Government insistence that Contractor F rely on the ad hoc system as a design baseline discouraged contractor innovation.
- The SR demonstrated very little application functionality, and it did not incorporate integration of applications. It therefore did not function as a mechanism to give stakeholders a shared understanding of the design.
- The SR was not shared with field users except at a limited number of demonstrations. User F and the Air Force SPO both felt that the risk of negative feedback on the program was greater than the benefit of end user feedback on a partial design.

Using the ad hoc system as a design baseline had both an advantage and a disadvantage. It leveraged past input from users, as captured by the ad hoc system, to specify design details that would otherwise have needed time and resources to determine. However, Contractor F created a new architecture and fresh implementation and

integration of capabilities, opening the door to possible innovations that might have enhanced operability for the user. The decision to depend on the ad hoc system seemed to diminish interest in seeking end user feedback on the design.

The strategy adopted by both services' user communities of relying on testers to incorporate user perspective into the design is difficult to evaluate given the information collected in this case study. While such involvement was probably beneficial, it is not clear to what degree field user perspectives were brought to bear by members of the test community. For any program, effectiveness of tester comments would depend to a large degree on the experience base of the testers.

Lessons learned for the program included the following:

- Reducing the number of IPT's and streamlining decision-making was important in allowing the program to regain it's footing after early programmatic difficulties.
- Since the contractor lacked exposure to the operational environment and user feedback was not available, the user SME was concerned that the system design might fail to consider some user operational considerations. If the program had been able share a partially functional SR with field users, it might have been possible to remedy much of this concern. However, this approach would have required devoting resources to create such a SR earlier and successfully managing field user expectation about the status of the design.

Contractor innovation on Program F was constrained both by government-enforced reliance on the ad hoc system as a design baseline, and by cost and schedule pressures. While the development software served as a SR for the program, it did not

reproduce critical aspects of the design, and field users had little interaction with it. User feedback on operational considerations associated with the design was therefore limited. Government-imposed design restrictions, programmatic pressures, SR limitations and minimal field user involvement may have contributed to the small number of valuable adaptations that emerged during Program F's design phase.

5.8 Case G

This case involved the design of a hardware and software suite to upgrade a major subsystem on a military aircraft. The R&D budget for the program was $140 million and the design phase spanned 24 months. The objectives of the program were to insert technology to replace obsolete hardware, create a pathway to periodically upgrade hardware using a robust range of suppliers, and minimize life cycle costs. Affordability and maintenance concerns were paramount, and user requirements were to maintain existing functionality. The approach was to re-host existing software functions onto Commercial Off the Shelf (COTS) hardware that would be made more rugged for a military environment. Because of COTS considerations, equipment reliability was also a high interest area for the stakeholders on the program.

Several categories of users were involved in the program. The end user was an operational wing that flew the aircraft that was being upgraded. The wing's primary interests were hardware maintenance and retaining existing functionality. An Air Force headquarters office managed formal requirements for the program, although the staff did not have operational experience with the aircraft. User perspectives were also introduced into this program by test organizations that had representatives on-site at the prime contractor facility. One of the test organizations (referred to here as User G) reported to the user headquarters mentioned above and had a staff with an operational background. User G was the primary user representative on site, and their personnel attended local system demonstrations and meetings to follow issues of user interest on the program. This case study relied on interviews with two members of User G to represent the user

perspective since these individuals provided the primary, day-to-day user interface to the contractor during design.

The System Representation (SR) for this program was a set of labs that contained representative hardware and development software in approximately the same configuration as was envisioned for the final system. The SR itself and the rationale for making this investment are described in the SR section below.

The program's design phase proceeded in segments. The system design was presented in preliminary and final form at two Technical Interchange Meetings (TIMs). Subsequent subsystem design efforts had three aspects: software design, aircraft infrastructure modifications, and new hardware design. Both the system level and subsystem level design phases were included in this study. The design phase ended just before the start of formal testing, when the aircraft modification design was completed and a technology insertion decision was made for the new hardware suite.

Software design involved minimal changes to existing functionality. The contractor strove to create the same look and feel of the legacy software so the new code would generate displays consistent with other aircraft system software. The primary difference was in the Built in Test (BIT) area, in which diagnostic software associated with new COTS hardware replaced legacy code, and BIT information was centralized to one location instead of being resident in separate pieces of equipment.

The software design team created weekly software builds for over 18 months. Due to the planned lack of functional differences, the user took little interest in software design, with the exception of the new BIT software. Contractor G noted that the user community gave valuable inputs on the BIT implementation, particularly regarding the

centralization of BIT data. However, the User subject matter expert (SME) voiced frustration that the COTS portion of the BIT software was already written and could not be modified. The implication for system adaptability was that most operational considerations of the software design were not issues for the user for Case G because the software either mirrored previous code or was pre-defined by a commercial vendor.

Several factors noted during the interviews affected program adaptability. Having User G on site meant the contractor could get a user perspective on aspects of the design more rapidly than would be possible through phone interactions or periodic meetings with Wing personnel. The SPO knew that rapidly changing technology would be a part of the equation during this effort, and they anticipated that unforeseen changes might add value to the program. They therefore made the case to set aside resources for studies and structured the contract to make such efforts possible, which made the task of adapting during contract execution much easier.

A third aspect of adaptability related to opportunity. The SPO SME indicated that the stakeholders discovered a unique class of changes –problems that had been identified in the legacy system that could be fixed easily. While technically not in the defined scope of work of the contract, many of these changes were inexpensive to implement because of the work that was already being done. Each such potential fix was treated as a collaborative decision between the SPO, the user and the contractor.

Because of the program's emphasis on reducing life cycle costs (LCC), the stakeholders benefited from another tool that helped with decisions about whether or not to adapt. The contractor developed a cost model that would quantify the LCC savings of a course of action on the program. This model was used to provide data to decision

makers regarding the value of potential adaptations. It was a valuable supplement to the labs, which provided only limited insight into LCC considerations. The tool functioned as a form of SR, supplying common information about the system to stakeholders, which reduced uncertainty and made it easier to reach consensus on potential adaptations.

The wing and user headquarters wanted to get the system upgrade to the field as soon as possible, and they made it clear that they did not want high-end computers with excess performance, but were more interested in reducing operating costs. Being on-site, user G was able to focus on the Case G development program, but they did not have a full and up-to-date operational perspective to the same degree as wing personnel. The SPO SME indicated the wing was very busy and often got diverted from paying attention to the contractor's' development activities.

The SPO subject matter expert (SME) indicated that the primary SPO emphasis on this program was on ensuring a strong understanding of program requirements as well as advocating and defending adequate resources to complete the program. He felt that defining the requirements thoroughly at the start helped achieve an on-time completion. The SPO also played a major technical role in evaluating the COTS hardware to ensure it would function reliably in the intended operating environment.

No major requirements shifts occurred during design, and the program was intentionally structured to avoid pushing technology limits. One major technology insertion option was structured early in the development effort, and the final decision to go with the lower technology, lower risk option was exercised just weeks before the start of formal testing. The SPO SME credits this decision with helping the program stay within cost and schedule constraints.

Contractor G had a broad experience with program G's aircraft from past development efforts, and they hired retired military personnel to obtain more operational experience. The SPO SME noted the contractor strengths including software timing and sizing analysis and modeling of how the software would act on the system. Contractor G emphasized their philosophy was to meet user expectations, making sure they had the right translation of requirements. They spent considerable time at the start of the contract planning with the SPO and the user community.

The User SME voiced some frustration that the contractor did not consider interoperability of the upgraded system with ground systems, and there appeared to be no contractual mechanism for the SPO to push interoperability issues. Contractor G had developed proprietary ground hardware, so the user felt they had no incentive to coordinate with experts on other ground systems.

The primary technical challenge on the program was ensuring the COTS hardware would work in a military environment. Such factors as robustness under shock and vibration and electromagnetic radiation and susceptibility of the COTS hardware had to be characterized and compared to the aircraft environment. This analysis determined acceptability of candidate COTS hardware as well as defining how the airplane infrastructure, including rack design and electromagnetic shielding, had to be modified. The SPO and user community undertook a major collaborative effort to evaluate the environmental requirements stated in the original specification for potential relaxation in order to make some of the COTS options viable.

Certain aspects of COTS maintenance required assessment by the stakeholders. In some cases, the maintenance approach called for Air Force maintainers to call vendors

for support instructions, which was a major change from relying on documentation. Another issue was the level of troubleshooting that was possible. The user wanted a capability to fix hardware down to the lowest possible level. COTS contractors typically established the level of repair at the circuit card level, which was one level higher than AF maintainers had experienced in the past. The vendor-supplied built in test (BIT) software was discussed above. These issues required discussion and evaluation with the user in order to ensure the COTS approach would be acceptable.

The SPO noted two substantive benefits to COTS - performance and life cycle cost reduction. At first the user wasn't interested in increased performance, but the SPO indicated users have found this aspect of COTS to be very advantageous. COTS hardware is easier and less expensive to upgrade than uniquely developed hardware. The SPO SME described that commercial COTS facilitates several small, incremental upgrades in an evolutionary fashion, versus the traditional military standard development involving a longer development period to achieve one big upgrade.

SR Description

The SR was described by the SPO as a family of labs at the contractor facility that were used to represent the system at the subsystem level. The labs were linkable to create a high fidelity system level effect. The result was a set of mission representative hardware and software that reproduced the design as it was envisioned at any given point in the design process. Contractor G's lab manager indicated the lab's primary purposes were to support training, integration, and testing. Also, the lab allowed the contractor to try new technologies and demonstrate the design for government feedback. For the Case

G upgrade, the lab was intentionally constructed in an office type environment, which was considerably cheaper than duplicating the aircraft configuration. The contractor viewed the labs as a worthwhile investment, both as a sophisticated development tool to narrow design choices and as a means of risk reduction.

The contractor had one software lab and hardware labs in three distinct areas. The technology insertion area was used to check out hardware performance and compatibility, including architectural selections early in the program. After the hardware baseline was established, this area was used to bring in next generation hardware to characterize the work scope that would be required to insert new equipment in the future. The second lab area was for development and integration. It was used for lower level tests, and often employed simulators instead of real devices. It was controlled informally to allow rapid changes in configurations to support internal contractor testing. The third area was called system integration and test. This area had highly representative hardware for the Case G subsystem, and could be patched to other subsystems such as communications for the purpose of end-to-end integration and testing activities. This area was used for a wide range of tests including unit level, software, subsystem and full-up system level testing. Changes to the integration and test area were tightly controlled to minimize configuration problems.

All three lab environments could be tied together by patch panels to simulate a variety of configurations. Sources of data could also be simulated with a high degree of fidelity to add greater system-to-system realism. As an example, equipment was available to broadcast and receive RF emissions as would be done with the real system.

SR Usage

The contractor lab manager indicated that the SPO saw SR demonstrations over the course of many TIMs but did not tend to interact with the lab equipment. User G personnel were in the lab every day during much of the development, typically focusing on how operators would use the system once it was delivered. Wing users were not as involved, primarily attending formal test activities. At times the contractor had to get the users out of the lab so they could focus on the work at hand. The contractor lab manager indicated this approach worked well since everyone understood the need to keep cost and schedule of the program on track.

Since users were on-site and involved in ongoing status meetings, they had an awareness of the state of the design and were not overly critical at early stages when the design was still maturing. Through interaction with the lab, User G representatives lived through the whole migration to the new design.

The contractor demonstrated system capabilities at every TIM and indicated any design trades that were under consideration. For example, they showed how the BIT would work at one TIM. A demonstration could last for 10 minutes or an hour. All three stakeholders noted that this approach was much more helpful than viewgraphs in imparting a detailed understanding of the system design.

The labs made it possible for the contractor to assess prototype hardware to provide decision-makers information on design options, including performance levels and work effort required for implementation. The contractor also used the labs to wring out the system, which provided risk reduction before installation on the aircraft. The general testing flow started with the technology lab, went to a separate software lab, and then

proceeded to the integration lab where the hardware and software would be utilized together in a full system configuration. According to the lab manager, they ran the hardware and software baseline and showed that it was bug-free well before flight test. They also demonstrated interoperability of system components.

Stakeholder Interactions

Several strong management practices helped the program G stakeholders achieve alignment. The SPO and contractor put a substantial amount of energy into defining a program Integrated Management Plan (IMP) and Integrated Management Schedule (IMS), which were contractual agreements. The IMP set entry and exit criteria for a series of program events that drove the progress of the program. These included Technical Interchange Meetings (TIMs) that served the function of formal design reviews. Both the SPO and contractor SME emphasized that the work done on the front end of the effort to define the IMP was a major factor in keeping the program cost and schedule on track. Another critical aspect of the IMP was that it required user feedback on every milestone before closeout could be approved. User G noted that the contractor consulted them for "everything" and sometimes was held up waiting for user feedback.

By making some of the program practices less formal, the SPO and Contractor G were able to speed information flow and reduce program costs. The SPO tailored the contract's formal data delivery list to a small fraction of the originally proposed set of documents, and documents not on the data list were made available upon request. Having TIMs instead of formal design reviews also cut down on some of the effort required on more traditional contracts.

Another management practice that was widespread for Program G stakeholders was inviting contractor, SPO and user representatives to attend all formally established periodic meetings. Several contractor meetings, including their systems engineering management team and integrated product teams (IPTs), had government representatives. The contractor indicated that User G participation varied with personalities, but they were always invited. All three stakeholders were involved in joint risk management meetings and weekly status teleconferences, and the user sat on the Configuration Control Board (CCB) to vote on any requirement baseline changes. This practice of open communication enhanced stakeholder understanding of program issues and activities, and the contractor SME felt it was pivotal in building trust between the parties.

The User SME commented that he saw many junior officers representing the SPO who did not have operational experience. He noted a tendency on the part of the SPO to make decisions without going to the user. One example involved a potential strategy of using circuit cards from redundant racks of equipment as part of the sparing scheme. The SPO did not work the issue because they didn't see an operator need, and user representatives had to explain the value of such an approach. This observation highlights the importance of having user personnel involved in operationally related design decisions to add perspective that most SPO personnel did not have.

The user SME noted that when user needs were uncovered that were not explicitly spelled out in the contract, they sometimes could not be met due to lack of funding. However, he noted that in other instances the contractor did the work if they knew the system was going to be unacceptable to the user without the change. Given that the program met its cost objectives, it can be inferred that SPO and contractor managers were

able to disposition such unexpected issues, deferring additional effort unless it fit within fiscal reserves or additional funds could be made available. The SPO SME reinforced that both the SPO and contractor put a high emphasis on cost control even though it was a cost reimbursable contract. The contractor SME indicated that unexpected requirements changes were handled by open discussion with the SPO to determine how they should be dispositioned.

Attitudes about trust on the program varied between the stakeholders. The user SME observed friction between the contractor and the SPO, particularly regarding contractor estimates for work. He felt that the contractor was more candid in informal discussions than at formal management meetings, where they seemed focused on projecting good news about the program. The contractor SME felt that the government had tremendous trust in the contractor because they had a record of delivering on promises. He felt that contractor emphasis on open and honest communications paid tremendous dividends in the form of government trust. The SPO SME related that the SPO-contractor partnership was unusually strong on this program, enabling a rare degree of close collaboration that substantively benefited the program.

The user SME felt that user involvement was different for requirements definition, where the user's predominant role was clearly understood, than it was during design, when involvement was "hit and miss." He commented that much of the interaction during design was "down in the weeds" below the level of boards or formal approvals. User G personnel frequently interacted directly with contractor technical personnel, and they would often tell an engineer about a change and get it implemented right away. The user SME described this interaction as having an enormous effect. If a

change was going to bust a deadline or revise drawings, it would typically be delayed to consider whether the contract needed to be modified.

Another user observation was that the contractor was sometimes more worried about the technology than what it was being used for. The user SME recalled instances in which the contractor lacked operational insight. As an example, contractor engineers originally designed a ground interface in which information equivalent to 150 emails an hour was being passed. This rate of data flow was not workable for operators. In such instances, the User SME felt the design was not being well thought out, and user interaction to provide an operational perspective was valuable.

The program's effectiveness at communication was a recurring observation for all three stakeholders in the interviews. Some of the key aspects of communications for the program are described below.

- The contractor SME recalled having no concerns about getting feedback in time to feed decisions, whether on a day-to-day basis or even more effectively due to instant feedback at TIMs.
- The contractor established a weekly list of issues and tasks, which provided a mechanism for discussing priorities and commitments. The contractor SME felt this practice was very important for aligning expectations.
- The program established a practice in which briefings were created jointly by the SPO and the contractor. One party would start a draft, and both would review it to get it done rapidly. The contractor recalled that this practice led to faster preparation times and less chart critiquing. Everyone realized the charts were a

work in progress, which was very comforting. Working together in this manner led to less criticism and more in-depth mutual understanding.

- Another contractor practice, encouraged by the program manager, was to ensure meeting notes were coordinated to agree on what commitments were made. It was important to capture the dynamics of different stakeholder positions if something changed. This practice helped keep everyone up to speed and allowed the parties to establish priorities. The contractor SME felt this open, real time coordination was a major factor in building government trust in the contractor.
- The user also noted that the contractor put a lot of emphasis on pre-staffing positions before decision meetings.

There were some minor communication challenges for the program. The user SME described that the formal decision process for the wing took a lot of time because users were so busy with flying, business trips and answering emails that they were operating under "information overload." Some user decisions could be delivered less formally in a rapid fashion. Another challenge that the SPO SME admitted to was that too many SPO engineers occasionally deluged the contractor, and that a balance had to be found to avoid overwhelming the contractor. In spite of these observations, the stakeholders provided a strong, consistent indication that full and open communication, and the trust that resulted, were major strengths of the program.

Adaptability

Table 5.7 provides a breakout of the number of collaborative changes of high, medium and low value (as defined in Table 4.6) for the program.

Number of collaborative changes	23
High value changes	2
Medium value changes	15
Low value changes	6

Table 5.7. Collaborative changes for case G

The collaborative changes fell into the following categories: two concerned the user interface, two related to maintenance, two impacted development cost, nine reduced life cycle cost and eight enhanced reliability.

The following is a partial list of Program G's collaborative changes:

- Selection of a display with inexpensive production cost to minimize LCC.
- Merging of software baselines between lab and operational software to reduce maintenance and training costs.
- Definition of hot (powered on) sparing process for change-over in the event of equipment failure.
- Resolution of design issues to ensure COTS compliance with environmental requirements.
- Selection of hardware for technology insertion to reduce development cost and program risk.
- Repackaging of some COTS hardware to improve performance and reliability.

- Agreement on requirement change to allow hardware failure in the event of catastrophic cabin depressurization prevented costly redesign of COTS hardware.
- Integration of power conditioning unit in another subsystem reduced the size of the equipment suite, lowering life cycle costs.

Program G was able to leverage off of a very high fidelity SR, which could be configured at subsystem or system levels, to share design information to the stakeholders through most of the design phase. Collaboration between SPO, user and contractor was very strong, as evidenced by the large number of positive interview comments about communication and trust between the stakeholders. These factors allowed several substantive collaborative changes to be identified, evaluated and implemented to the benefit of the program.

Within the context of the eight case studies performed for this research, program G was considered highly adaptable, falling in the second highest tier of adaptability. As described in section 6.2, two of the cases exhibited more adaptability, four exhibited less adaptability and one other case was also rated as highly adaptable.

Summary

Program G stakeholders used an array of strong communication and collaboration practices to build shared understanding and keep all parties aligned. This alignment led to enhanced trust and continued open sharing of information. Forging agreement on the IMP and IMS gave structure to the program and was a major part of this success. Informal data and review practices also helped. All stakeholders were invited to a wide

range of recurring program meetings, and the contractor captured agreements and changes in stakeholder positions in writing. These practices provided a solid foundation of stakeholder interaction that was leveraged to address program challenges and potential adaptations in a highly collaborative manner. Certain other practices and program circumstances also helped the program adapt, as described in the lessons learned below.

Both the SR and the contractor's cost model tool for understanding decisions impacting life cycle cost considerations provided mechanisms for sharing information, identifying potential adaptations and evaluating those adaptations. The SR was configurable at the system level, included high fidelity simulation of interfaces, and provided insight into such major emphasis areas as adequacy of performance, maintenance considerations, and reliability of COTS hardware. The cost model added insight into the last major area of interest – life cycle cost reduction. These tools greatly facilitated program adaptability by enabling knowledge sharing between stakeholders and in-depth exploration of design options.

The following lessons learned were observed for this program.

- COTS hardware posed maintenance and reliability concerns, but provided reduced LCC and opportunities for rapid performance upgrades.
- Having money and a contract mechanism for studies helped adaptability.
- The LCC cost model tool filled in a gap that the SR could not portray, providing shared insight to stakeholders in support of LCC-related adaptation decisions.
- Testers on-site provided a means to inject operational considerations into the design.
- Identifying legacy problems for inclusion during the early design effort (when cost was minimal) added value to the final product.

- None of the stakeholders had responsibility for assessment of interoperability options with ground systems, and contract incentives were not established to encourage the contractor to address this area of system development. As a result, the overall program may have incurred a risk that redesign might be needed in the future to address ground system interfaces.
- The cost for producing the labs was justifiable due to their use as a development tool and as a means of reducing program risk. The labs became a part of the overall program infrastructure, reducing long term LCC.

Due to the factors described above, the stakeholders for Program G were able to resolve several issues that were significant to the program in a highly collaborative manner. COTS concerns were overcome, a technology insertion option was held until late in the program without perturbing program constraints, and unexpected requirements that emerged during design were dispositioned effectively. Also, a large number of valuable collaborative changes were implemented. Strong collaboration, coupled with a high fidelity SR and LCC cost model, made this program highly adaptive.

5.9 Case H

Program H was an Air Force hardware and software development to add a new communications subsystem to an existing military aircraft. A previous version of this communication capability, referred to here as the “contingency system”, had been assembled rapidly in an ad hoc manner and added to aircraft in support of a military operation. After the operation, the equipment remained on the aircraft. However, the contingency system had several deficiencies, including performance shortfalls and support issues that the user wanted to be corrected. Program H involved a complete technology change from the contingency effort. The R&D budget for the program was \$40 million, and the design phase lasted for 24 months. User requirements called for the same general capabilities as the contingency system, but with expanded interoperability with other systems. This case addresses the Program H design phase through it’s first critical design review (CDR) with the original equipment baseline.

Four categories of users were involved in Program H. The first category involved user headquarters offices that coordinated requirements for the aircraft system. The SPO SME noted that one of these offices (“User H”) integrated user requirements. Secondly, the Wing that flew the aircraft provided an operational viewpoint. The third category included test personnel co-located with the contractor who sometimes provided a user perspective during the development. The U.S. Army represented a fourth category of user, since they needed mission data from the aircraft that was to be provided by the new communication capability. This case study obtained user perspective from an individual (referred to here as the user SME) who had past experience with Program H and the contingency system while at the Wing and at one of the headquarters offices.

Program requirements were documented by the SPO in a Technical Requirements Document (TRD) that became part of the contract. The SPO SME indicated that the user headquarters and the wing coordinated on the TRD and added some requirements, but the SPO played the primary role in defining requirements in the pre-contract award period. After contract award, the Wing levied several major new requirements that specified ways to use the system in conjunction with the contingency system and other airborne and ground users. Funding was provided for a contract change, but the SPO SME commented that schedule pressures led to disconnects in understanding and a poor capture of cost implications. The SPO recalled engaging in a series of iterative discussions with the user headquarters, the Wing and Contractor H to clarify specifics of user needs and the contractor's approach to achieve them.

The SPO and the user experienced benefits from the existence of the operational contingency system. The SPO used the contingency system's capabilities as a starting point for documenting a technical baseline in the TRD. The Wing was able to see what they liked and what was missing from the contingency system, which helped define expectations for the Program H development.

One central element of the new system was a radio that was provided by another developer as government furnished equipment (GFE) to contractor H. Recurring delays in the delivery of the radio proved to be a major complication, driving numerous schedule re-baselines. The SPO SME felt that Contractor H had been very responsible in shouldering the burden caused by the late hardware. Contractor H was careful to keep the user informed of how the radio's design was impacting system capabilities. The contractor SME characterized the situation as an ongoing effort to figure out how the user

could live within constraints imposed by the radio and still do the best job of satisfying user needs. The implications for the design stemming from non-availability of the radio included inability to test software against the radio and limitations in the fidelity of the representative labs that acted as a system representation (SR) for the program (see SR description section below).

All three stakeholders emphasized that Program H was very highly constrained. Data had to meet a network timing protocol, Department of Defense (DoD) architecture standards, and interface standards for compatibility with other systems. The system had to support frequency requirements in a wide range of geographic areas. Also, the design of the GFE radio constrained what the system could do. The SPO SME stated that many issues were political, with different parties requiring equipment to be part of the system architecture and adding interoperability requirements. These constraints led to a very complex set of technical requirements and very tight funding for the program.

The SPO played a strong systems engineering role in the program. Their focus included ensuring all cost and technical constraints were being met, trying to optimize the data flow through the limited capacity of the system, and balancing the data needs of a diverse set of users. They also expended considerable effort investigating potential interference between antennas to make sure the new system would function reliably in the aircraft environment. The SPO SME commented that many novel aspects of the system were coming together at the same time, which required the SPO to take a systems view. He felt the complexity of the system was not fully appreciated by the user community, some of whom seemed to feel the effort was little more than adding a new radio to the aircraft. System capabilities sprang from both the radio and implementation

choices in the software design. Given these considerations, the primary areas of emphasis for the SPO were achieving the required capabilities, enhancing interoperability in support of users and ensuring the system was reliable.

The SPO SME noted some of Contractor H's contributions to the program. He recalled that they had understood schedule implications of other systems in development on the aircraft and were very active in proposing solutions to interoperability issues between these systems. Also, the contractor was able to provide evidence that simulation could replace some testing, resulting in cost savings. The user SME felt that the contractor was very responsive to user needs, although he recalled that financial pressures grew on the program, leading to more requests for additional compensation in response to user issues.

Some factors raised in the interviews seemed to be acting to limit adaptability on the program. The SPO felt that Contractor H had underestimated the work effort and noted that budget instability added further financial strains. Given cost constraints, the contractor was less likely to delve into potential innovations. The SPO SME also indicated that requirements had been trimmed to the bone, making each new issue or constraint very painful to explore and fit in. He stated that the contractor and SPO were going through such exercises "perpetually." The late GFE radio, because of its shifting capability baseline, was another limiting factor for exploration of program options.

On the positive side, user experience from the contingency system and detailed discussions to resolve issues and constraints led to several adaptations for the program. As new requirements or constraints were identified, the contractor would determine what they could do within budget constraints or the program would seek additional funding.

Many of the challenges for the program involved data flow issues. The contractor built a model of the communications system, but the SPO noted that funding was very limited for model development. The SPO performed independent calculations and found disparities with the model, so it never emerged as a key tool for the program. Instead, the SPO SME indicated that the SPO and contractor would have detailed discussions to understand and resolve issues. He indicated that he had to hold a web of details in his mind to understand the system, which seemed to imply that there was no easy way to represent the system data flows that could be shared between parties.

The SPO SME felt that it was often difficult for users to specify in writing the full extent of their needs, and they were too busy to keep deeply involved as the design progressed. Because of the late GFE radio there was no tool to represent the full system design as it was unfolding. The contractor SME recalled that user emphasis was primarily on performance (time for the system to do tasks), link quality, and how the system would be operated.

Another problem faced by the program was the lack of a detailed concept of how the system would be used. This information, which falls in between requirements and design detail, is referred to through this research as operational steps. The SPO SME indicated that the contractor had to speculate on how the system would be used to figure out how to implement the design. He described this as more an art than a science. The SPO recalled a great deal of discussion between the SPO, Contractor H, the Wing and the Army to describe how much the aircraft and ground personnel would be talking. Sometimes this became a political issue, but the results would impact design and performance of the system.

The SPO SME felt the user community didn't understand the importance of defining how the operator would use the system. He felt the user knew it was very complicated and they didn't know how to capture it in a written description. Users were busy and did not seem to have a clear sense of their role in the design process. The SPO SME perceived that users were waiting for tools to help assess the design, and absent that, they delegated the role of monitoring design progress to the SPO. The SPO noted that it was hard to find time to grow an understanding of operational steps in a group. As the program progressed, the contractor's design team had to make choices. The SPO SME thought it would have been better if the participants had a way to discuss such choices in a more collaborative and comprehensive way.

SR Description

The system representation (SR) for Program H was the same representative lab described in Case G. The biggest difference in the case of Program H was that, during the design phase, the GFE radio was not available for integration into the lab. This factor limited the utility of the lab as a representation of the overall design. Both the software design and the non-radio hardware modifications to the aircraft were represented in the lab.

Periodically, early prototypes of the GFE radio were used in the lab, but they had limited functionality to the point that it was not possible to perform a true system-level demonstration.

SR Usage

Interviews revealed three different forums in which the SR was used on the program. At Technical Interchange Meetings (TIMs), the contractor would show the current system configuration in the lab and highlight projected changes. They would explain what they were on contract to do and go over the timeline. The contractor also did demonstrations of interim software releases in the lab with the SPO and user representatives on an informal basis. The contractor SME recalled getting some SPO feedback at these sessions. The third forum for the SR was at the two design reviews.

Design review demonstrations were done with early versions of the GFE radio, whose poor functionality made it impractical for the users to interact with the system and give meaningful feedback. The contractor recalled that the user was able to consider maintainability aspects of the hardware in the lab. However, the main area of discussion was the human – machine interface, including analysis of the screens.

The user SME recalled that at the final design review, the demonstration setup resulted in discovery of a concern with how the radios could be operated if the main computer was not on, which happened during much of the time the aircraft was on a mission. The SPO had assumed the user would not need the new Program H communications capability in this circumstance, but discussions revealed that this was an important capability. As a result of this discovery, the contractor included additional hardware in the design to allow activation of the radios without the central computer. The user SME felt that this issue would not have been discovered in a briefing of the design, but was visible because of the SR.

Another benefit of the SR was that, by providing a physical duplicate of the aircraft, it allowed the contractor to resolve installation issues in advance.

The SPO SME discussed some of the limitations of the lab. He felt that financial pressures and the lack of the GFE radio made it impractical to use the SR to portray the entire system so that it could be assessed holistically. The SPO SME felt that having a full system-level representation might have surfaced additional issues with the operational use of the system, but he could not say to what extent this might have been the case.

Stakeholder Interactions

The SPO SME felt that User H was effective during the design phase at integrating requirements and establishing official positions for the user community. Even though the user headquarters offices were thinly staffed, the SPO felt they were responsive on important issues. In the case of new requirements, the SPO recalled that the user would often take a position instead of collaborating to find a mutually agreeable solution. The SPO had to remind the user that excess costs could get the program cancelled in order to get them to collaborate on reaching a compromise.

The SPO, Contractor H, the Wing, and user headquarters personnel engaged in detailed discussions at a series of user working group meetings. The SPO SME stated that having everyone in the same room was very helpful for discussing and resolving design issues. During many of these meetings, the contractor showed static images of system screens and discussed process steps of the operators to provide insight into the

evolving design. The User SME recalled that there were no scripted sequences to portray how the operator would navigate between screens. Still, he felt the working groups were very beneficial. According to the contractor SME, user explanations of how they would operate the system were valuable complements to the specification and to SPO guidance. As an example, the contractor needed to make decisions on which radio capabilities would be enabled and disabled by System H software to create an efficient design. User involvement was critical in making these decisions. Even with the working group meetings as a mechanism for collaboration, the contractor noted that there were hundreds of action items at the final design review. This observation raises the possibility that methods such as discussions and briefings may have had limited effectiveness at sharing enough information to surface issues.

The contractor SME recalled that program planning meetings were painful and confrontational. He felt the work scope of the program was not difficult, but the ongoing delays in the GFE radio schedule forced a series of re-planning exercises that tended to be more contentious and less productive than the program's technical meetings.

SPO influence on design was substantial. The contractor stated that they proposed the design that the SPO had defined. The contractor SME recalled that a recommendation to move processing functions from one box to another that was contested with the SPO for six months because of other SPO priorities. The SPO took a major role in evaluating technical considerations such as potential interference between aircraft antennas, in which they had strong expertise. However, both the contractor and the user remarked that the SPO lacked a detailed understanding of how the user would operate the system.

Tensions and mistrust between the SPO and the user community seemed to stem in part from a limited understanding of each other's knowledge about different aspects of the program. The user SME had the impression that the SPO was focused more on managing program constraints than fulfilling user needs. However, both the SPO and the contractor commented that the user community underestimated the complexity of the program and ceded a great deal of control to the SPO for design-level decisions.

The SPO SME felt that they got the gist of user needs and passed this insight on to the contractor. He indicated they tried to gain the user context by getting as much feedback as they could from the user community. Users were required to sign off on the content of design TIMs, formal working groups, and all formal milestones. Sometimes issues dragged out, and in cases where the SPO was unable to get information or a position from users, they had to make a decision themselves. It was critical to have an answer so the contractor could move on with the program. The SPO SME stated that in such cases the SPO had to rely on their own experience, intuition, and technical expertise.

The user SME felt the user community had not had much influence on the program design. However, the contractor SME recalled the user being engaged, particularly in the areas of the human interface and hardware maintenance. This involvement allowed the contractor to focus on deeper issues in the design, which he felt was essential. The contractor recalled holding off on design decisions on several occasions until they could get user feedback.

The contractor relationship with the user was complicated, in part because of SPO insistence on being an intermediary. There was tension between the contractor's desire for clarification of user needs and the SPO's concern for cost growth in the absence of

adequate oversight. The User SME noted that the contractor would draw on informal interaction with local test organizations to get the user's perspective since Wing personnel were typically busy and hard to reach. However, the testers did not have operational experience with the aircraft and its systems. Another avenue employed occasionally by the contractor was contact through field representatives at the Wing. The contractor SME indicated that these individuals were typically asking questions of clarification below the level of formal requirements. However, this dialog became a concern for the SPO. The user SME saw that this interaction could sometimes go too far when a "gut feeling" from a user might result in cost impacts such as drawing changes.

The SPO SME described some concerns with the contractor. First, he felt they were not open in sharing information on program issues. They often delayed reporting problems and missed deadlines such as design TIM data package deliveries. Contractor bonuses were tied to SPO feedback, which seemed to make them very sensitive to any negative perceptions from the SPO. Second, they would sometimes bypass the SPO and market new capabilities to users, which could lead to new program requirements. The SPO felt the contractor should have come to them to describe any new technology opportunities, and also the user should have come to the SPO with any changes in their needs. The third concern related to contractor cost estimates. The SPO SME remembered receiving contractor assurances that design choices would not have much cost impact, only to learn later that large costs were involved. The SPO had to do its own cost estimates as a check and balance. A fourth issue concerned the contractor's tendency to argue over details of the scope of work. The SPO SME felt that some of these cases involved clear-cut interpretations of the specification. Many of these issues

were exacerbated by tight funding on the program and differences of opinion between SPO and contractor personnel.

Adaptability

Table 5.8 provides a breakout of the number of collaborative changes of high, medium and low value (as defined in Table 4.6) for the program.

Number of collaborative changes	17
High value changes	5
Medium value changes	11
Low value changes	1

Table 5.8. Collaborative changes for case H

The collaborative changes fell into categories as follows: two concerned technical performance, two involved the user interface, nine related to interoperability, one had to do with life cycle costs, and 3 impacted reliability.

The following is a partial list of Program H's collaborative changes:

- Added capability for separate, simultaneous data streams for the two radios.
- Added capability to be backward compatible with the contingency communication system.
- Added and defined additional message content that would flow through the system.
- Defined requirements and measures for data sharing by network participants.

- Selected an alternate source for data protocol to enhance interoperability.
- Defined a means for the communication system to be used when the aircraft main computer was off during a mission.
- Defined buffers and queue management to effectively manage data from multiple participants in the communications network.
- Defined format of free text messages.
- Added capability to relay selected messages using an additional network.

Many of Program H's adaptations came because of late user requirements that could have been specified and funded before contract award. Other changes were imposed on the program as additional constraints were defined. Adaptations therefore often came from necessity as well as from collaboration. However, collaboration, particularly between the SPO and the contractor, was important to specify how the program would respond to meet the new challenges while either staying within cost constraints or seeking additional funding.

Within the pool of eight case studies performed for this research, program H was considered highly adaptable, falling in the second highest tier of adaptability. As described in section 6.2, two of the cases exhibited more adaptability, four exhibited less adaptability and one other case was also rated as highly adaptable.

Summary

While this program exhibited adaptability in response to several challenges, a number of factors limited collaboration and knowledge sharing. SPO attempts to involve

users in the requirements process met with limited success, partly because of user perceptions that the system represented a simple change in capability. During design, user representatives participated in discussions at working groups and interacted informally with the contractor, but the primary operational community (the Wing) was too busy to make a major contribution. The contractor SME felt that the program was established without proper funding or a fully resolved technical baseline, which made program execution difficult. Several requirements and issues surfaced after contract award. Also, lack of insight into the GFE radio capabilities was a disruptive factor.

Program H provided the following lessons learned:

- While the SPO attempted to draw user representatives into the requirements process before contract award, a number of significant requirements were identified later. More substantive involvement by the user community might have permitted the program to start with a stronger consensus on the technical baseline.
- The parties underestimated the effort required to define operational steps that the users would employ to operate the system. Since user resources were limited, the contractor had to make some design decisions related to operational characteristics of the system without user input. The SPO, user and contractor need to plan for and dedicate resources to definition of operational steps so system designers will have appropriate and timely insight into operational considerations.
- The program put a strong emphasis on interoperability, but it did not possess a tool to represent interoperability considerations to all stakeholders. The contractor attempted to create a model, but did not have sufficient resources. Given the importance of understanding data flow considerations, the program may have benefited from a

mechanism to allow all stakeholders to understand the data flow as the design effort was progressing. For Program H, the absence of design details on the GFE radio would have made such an approach more difficult to implement.

- The user community seemed to lack a strong sense of their role and importance during the design phase. Providing user representatives with a means to digest the evolving design and a clearly defined role in supplying operational considerations for designers might have given them incentive to be more deeply involved.
- Stakeholders agreed that user working groups had a positive impact on the program by providing user input into the design process. However, the number of action items at the final design review was an indicator that the discussions and briefings done on the program were not touching on all contentious aspects of the design.

Stakeholders were focused to a large degree on clarifying and responding to shifting system requirements and complications arising from the late GFE radio. Although collaboration between the SPO and user was restrained and tense, and contractor interaction with the user was limited, the stakeholders were able to accommodate many changes through detailed discussions and contract modifications. However, the program was less effective at capturing user feedback on operational considerations as the design matured.

5.10 Summary of Cases

Table 5.9 provides summary matrices of the major characteristics of the eight programs. The following areas are addressed:

- Adaptive strengths – summary of key program characteristics enhancing adaptability
- Adaptive weaknesses – summary of key program characteristics impacting adaptability
- R&D $ (million) – research and development budget
- Design (months) – duration of design phase
- SR – description of SR
- SR Usage Level - level of knowledge sharing using the SR: exceptional, strong, moderate, weak or none
- System-level depiction – level of SR depiction: system, subsystem or minimal detail
- Adaptability – level of adaptability: very high, high, moderate, or low
- #CC- H/M/L – number of collaborative changes, and number of high, medium and low changes

	Case A	Case B	Case C	Case D
Adaptive strengths	-User & contractor co-located -User technical expertise -SR used operationally	-SR at SPO and contractor facilities -Open communication - Shared objectives	-Prototype gave baseline for req'ts & design -Built up-front consensus on design	-SR helped resolve issues -Changes anticipated and welcomed
Adaptive weaknesses	None	-4 users -Lacked stable concept of ops.	-No SR access -Remote users -$ constrained -No formal concept of ops.	-Req'ts not well developed -$ constrained -4 users
R&D ($ mil.)	24	23	13	22 (appx.)
Design (mos.)	43	15	14	16
SR	Development system	Development software	Fielded prototype	Development software
SR Usage level	Exceptional	Moderate	None	Moderate
System-Level Depiction	System	System	Minimal	Subsystem
Adaptability	Very high	Very high	Moderate	Moderate
#CC- H/M/L	19- 12/6/1	34 – 4/6/24	8 – 5/3/0	9 – 5/4/0

	Case E	Case F	Case G	Case H
Adaptive strengths	-Informal user feedback -Contractor made changes informally	- SPO in touch with legacy system users	-On-site testers had ops experience -Planning for tech. insertion, changes -Life cycle cost decision model -Early planning	-Detailed issue discussions -Informal user feedback -Strong SPO systems engineering
Adaptive weaknesses	-$ constrained -Low priority -SPO and user friction -User HQ lack of operational experience	-Two SPOs -Long decision process (early) -No concept of operations -Job scope underestimated -Limited user involvement	-Field user busy	-Limited user interaction with contractor -Late req'ts -Late radio -Program highly constrained/complex -Job scope underestimated -No formal concept of ops.
R&D ($ mil.)	15 (appx.)	28 (appx.)	140	40
Design (mos.)	6	21	24	24
SR	Development software	Development software	Representative lab	Representative lab
SR Usage level	Weak	Weak	Strong	Weak
System-Level Depiction	Subsystem	Minimal	System	Subsystem
Adaptability	Moderate	Low	High	High
#CC- H/M/L	12 – 2/10/0	10 – 2/3/5	23 – 2/15/6	17 – 5/11/1

Table 5.9. Summary of programs A through H

Chapter 6 Analysis of Adaptability and System Representations

6.1 Introduction

The eight programs described in chapter 5 provide a diverse set of examples of collaborative practices and adaptability during the design phase of program development. This chapter covers analysis of these programs related to their adaptability and to their experiences with system representations.

The initial section of this chapter describes the thought process and data for comparing the eight programs in terms of their level of adaptability. Then, findings are developed related to the two research questions listed below.

- How does a system representation enhance adaptability?
- What characteristics make system representations effective at promoting adaptability?

The case studies revealed stakeholder practices that implied analysis (i.e. a quantitative calculation such as a life cycle cost analysis) could play a part in facilitating knowledge sharing and adaptability. The chapter continues with a discussion of the efficacy of a system representation versus an analysis in sharing knowledge given different stakeholder emphasis areas.

The chapter ends with a summary of findings and ties the findings to theoretical implications developed from CAS theory in chapter 3.

Chapter 7 will address additional research questions concerning stakeholder roles.

The findings of this research relate to Air Force command and control programs. These programs focused on communications and computer technology and software development, which tend to be characterized by faster technology cycles (introduction of new technologies) and more rapid development times than other technology sectors. While findings presented here may apply in other defense sectors such as aircraft or munitions, such applicability remains to be proven. Similarly, applicability of the findings to programs with smaller or larger research and development budgets than the sample size that was studied ($13 - $140 million) would require further research.

6.2 Program Adaptability

This section presents data and rationale for comparing the eight programs in terms of their adaptability. As described in chapter 4, data collection resulted in determination of a quantity of collaborative changes associated with each program and a quality level (high, medium or low value) associated with each collaborative change. Both quantity and quality of changes were of interest in assessing the merits of the programs. Quantity was an indicator of speed in adapting, and quality provided a sense of the value accrued to the programs due to their ability to adapt.

Two programs achieved superior results compared to the others. Program B had the largest quantity of collaborative changes. It totaled 34 changes compared to 23 for the next highest program. Program A had the most high value collaborative changes, achieving 12 compared to 5 or fewer for other programs. These differences set programs A and B apart from the other six. They were considered “very high” in adaptability

within the context of the pool of studied programs. Figure 6.1 shows the program totals for number of changes, and Figure 6.2 shows the number of high value changes.

Total Number of Collaborative Changes

Figure 6.1. Total number of collaborative changes (CCs)

Number of High Value Collaborative Changes

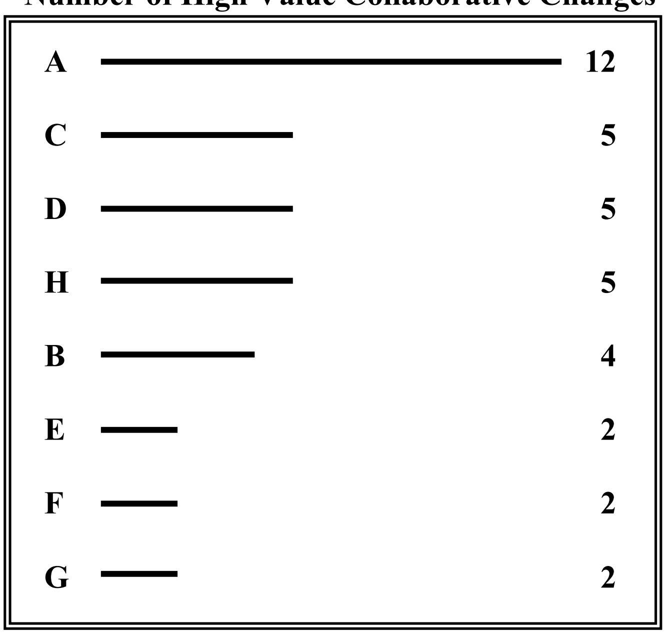


Figure 6.2. Total number of high value collaborative changes (CCs)

Differentiating between Program A and Program B was not practical since to do so would have required applying purely subjective weighting criteria to mesh quantity and quality factors together. However, differences in the nature of the two programs provide insight into the reasons they were able to stand out respectively in quantity and quality of changes.

Program B was a software development program. According to the stakeholder assessments of collaborative changes, both the risk of implementation and the value of 70% of the collaborative changes were low (the smallest values on five-point subjective scales). This assessment implies that these changes were not complex and were easy to evaluate and implement, facilitating rapid processing of a large quantity of changes.

By contrast, Program A involved design and integration of technologically advanced hardware and software. Stakeholders indicated that many of the collaborative changes for this program went through detailed evaluations since the potential risks and rewards for the changes were frequently substantial. Twelve of the observed changes for Program A were assessed as being of high value.

In summary, these programs demonstrated a very high level of adaptability in two different manners, reflecting their particular circumstances. Both attained results that set them apart from the other six programs.

The remaining six programs fell into two easily distinguishable groups based on the number of collaborative changes they achieved. Programs G and H were similar, with 23 and 17 changes respectively. These two programs were considered “high” in adaptability. The remaining programs (C, D, E and F) had between 8 and 12 changes.

However, half of Program F's changes were low value, compared to no low value changes for C, D and E.

The sum of high and medium value changes was chosen as the criterion to differentiate between these six programs. This criterion provided a means of accentuating the importance of the value of changes while still factoring in the relative quantity of changes. This approach yielded the same ranking as using the total number of collaborative changes, with the exception that it differentiated Program F as having a substantially lower cumulative value of changes than the other programs.

On the basis of this criterion, these six programs were sorted into three adaptability levels – "high" (G and H), "moderate" (E, D and C) and "low" (F.) Programs A and B were categorized as "very high" for adaptability, as described earlier. The program ranking, number and value of collaborative changes, and rationale for the ranking is summarized in Figure 6.3. Future references to program adaptability refer to the rankings established in this section.

Consideration was given to whether the number of collaborative changes should be normalized to reflect a program characteristic such as budget or design phase duration. Appendix B presents an analysis of program characteristics (requirements uncertainty, research and development budget and design phase duration) that determines there is no correlation between any of the characteristics and program adaptability levels. Therefore, no normalization of the number of collaborative changes was deemed to be necessary.

Program Adaptability Levels

Adaptability	Case	#CC	HIGH VALUE CCs	MEDIUM VALUE CCs	LOW VALUE CCs	Criteria for Ranking
Very High	B	34	4	6	24	Most #CC*
	A	19	12	6	1	Most HIGH VALUE*
High	G	23	2	15	6	17 HIGH+MEDIUM
	H	17	5	11	1	16 HIGH+MEDIUM
Moderate	E	12	2	10	0	12 HIGH+MEDIUM
	D	9	5	4	0	9 HIGH+MEDIUM
	C	8	5	3	0	8 HIGH+MEDIUM
Low	F	10	2	3	5	5 HIGH+MEDIUM

Key:

CC refers to "collaborative change"

*** Standout programs: exceeded all others in either quantity or quality of CCs**

Rationale for adaptability level

Figure 6.3. Program adaptability levels

In order to assess the validity of this ranking approach, a sensitivity analysis was performed. While this analysis is subjective, it provides an alternate way of looking at the data. Factors were chosen as multipliers to represent the relatively greater value of "medium" changes over "low" changes and "high" changes over "medium" changes. Figure 6.4 shows the results. The "base" series refers to the total number of changes for each program, and each "factor" series refers to a sum, or score, calculated as follows:

Adaptability Score = L + Factor*M + ($Factor^2$)*H

Explanation:
L = number of low quality collaborative changes,
M = number of medium quality collaborative changes,
H = number of high quality collaborative changes, and
Factor is a value of 1.5, 2.0, 2.5 or 3.0, depending on the series

For example, the series labeled "factor 1.5" assumes that medium value changes are 50% more valuable than low value changes, and high value changes are 50% more valuable than medium value changes.

This approach provides purely subjective comparisons, but several robust patterns emerged from the analysis. Program A became one of the top two programs for the factor 1.5 and higher series. Program B remained one of the top two cases unless a factor of greater than 3 was applied. At that point, Program H started to catch up due to its larger number of medium value changes. The relative positions of H and B could therefore be argued, depending on the choice of factors.

At the other end of the spectrum, Program F dropped to the bottom of the scoring when a factor of 1.5 or more was applied, and it's position became more pronounced for larger factors. This trend is due to the preponderance of Program F's low value changes.

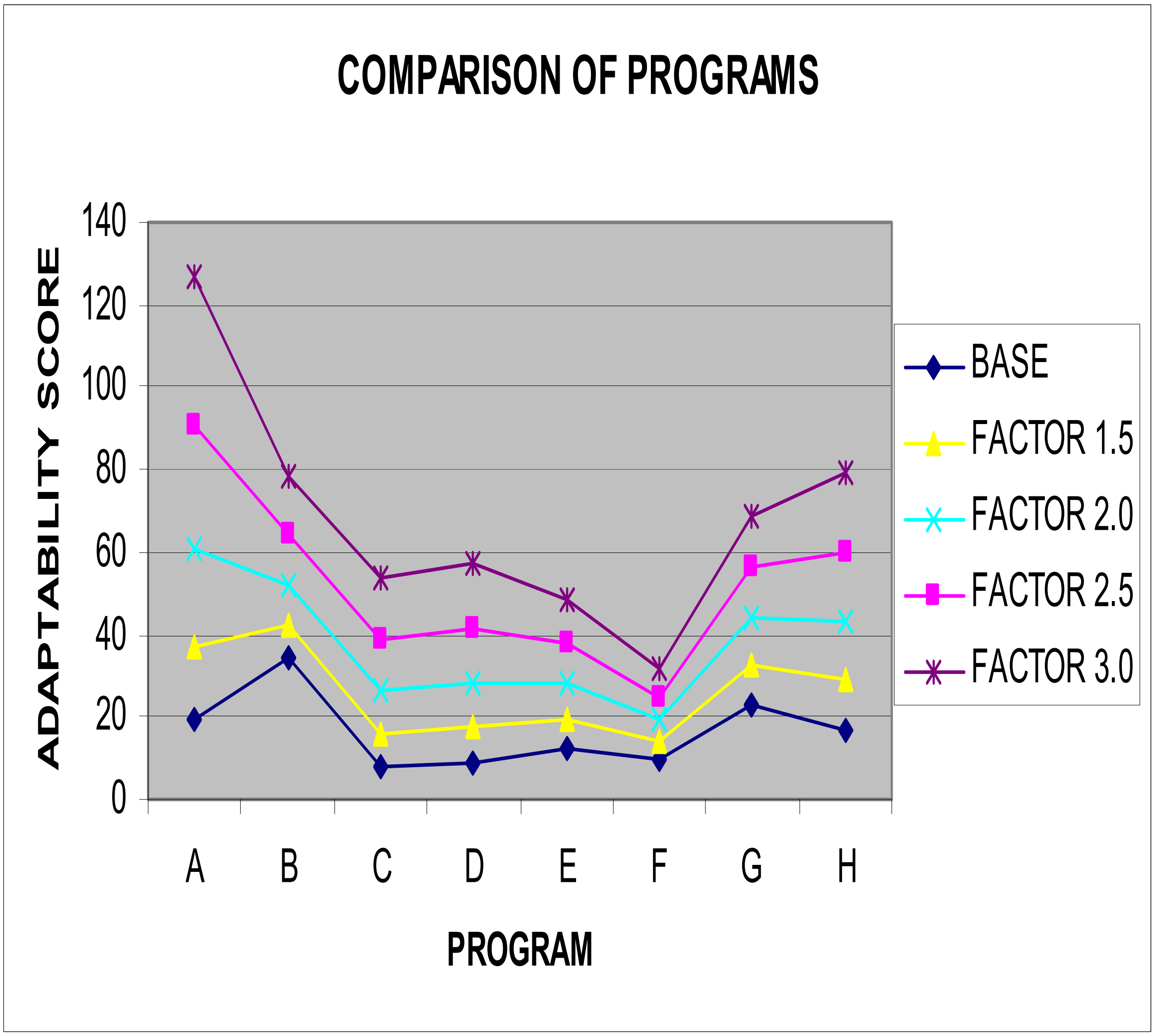


Figure 6.4. Sensitivity analysis of programs

Two other clusters of programs appeared. Programs G and H were similar, and they maintained a higher score over another cluster of programs - C, D and E.

The patterns revealed by the sensitivity analysis reinforce the validity of the relative program rankings presented in Figure 6.3, in which programs A and B are in the highest tier, G and H are in a second tier, C, D and E are in a third tier, and Program F is in a lowest tier. The potential juxtaposition of B and H was resolved in favor of B due to

the substantively larger number of total collaborative changes (34 versus 23) observed for Program B.

6.3 System Representations and Adaptability – Knowledge Sharing

This section addresses the research question: "How does a system representation enhance adaptability?" To approach this question, data from the interviews were evaluated to establish insights into the ways programs used system representations. The degree of knowledge sharing using system representations was investigated due to a potential influence on program adaptability. As part of this thought process, the frequency and depth of interaction with the SR was noted for each program.

To evaluate the significance for adaptability of knowledge sharing using a system representation, a multi-level criterion was developed to differentiate between the programs. The levels of the criterion are defined in Table 6.1.

Knowledge Sharing Level	Definition of Levels
Exceptional	In-depth stakeholder interaction (fully exercising the design using the SR) on a daily basis (majority of the design phase)
Strong	Substantive stakeholder interaction (exercising some aspects of the design with the SR) on a daily basis (majority of design phase)
Moderate	Substantive stakeholder interaction (exercising some aspects of the design with the SR) on a weekly basis or less (last quarter or more of design phase)
Weak	Top-level interaction (brief exposure to minimal aspects of the design through the SR) on a weekly basis or less (last quarter or more of design phase)
None	No interaction with a SR

Table 6.1. Levels of Knowledge Sharing

Program A's user had daily, substantive interaction with the SR. The user was co-located with the contractor and was operating the SR to produce data products for the field. User personnel were therefore intimately familiar with all aspects of the design of the system as it evolved. This level of knowledge sharing was "exceptional" as defined by the above criteria. It should be noted that Contractor A felt that the close level of collaboration between contractor and user on this program might not scale well to a program of larger size. The concern was that contractor managers would not have sufficient span of control to monitor the resources expended by a large number of employees if they were responsive to user queries.

Program G put in place a functionally representative set of labs that mimicked the evolving hardware and software design with a high degree of fidelity. On-site user

representatives were able to interact with the SR on a daily basis during most of the design phase, although their interaction was limited to a subset of design features at any given time. This level of knowledge transfer about the design of the system was "strong" per the established criteria.

Both Program B and Program D used a system representation to present design information to government stakeholders at periodic meetings. However, for both programs, not all aspects of the design were captured by the system representation until the end of the design phase. These programs met the "moderate" criteria for knowledge sharing.

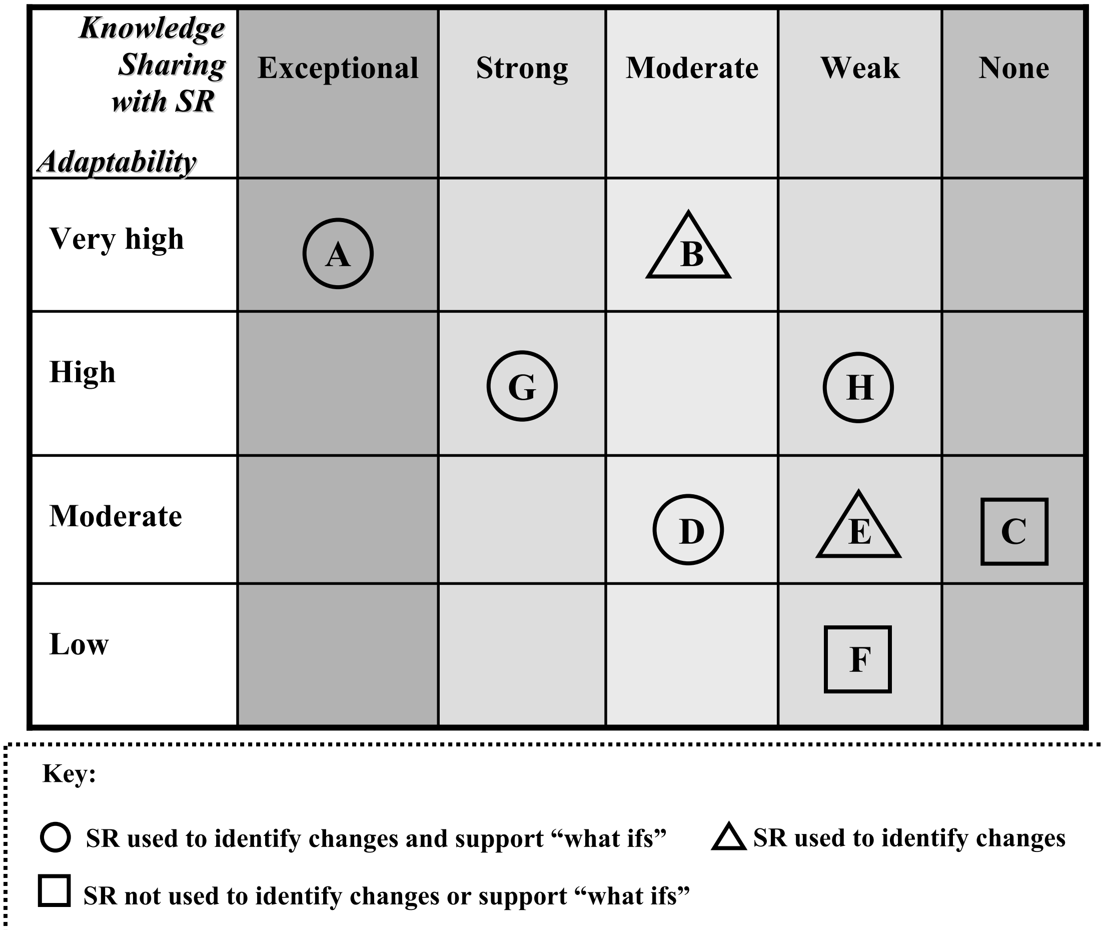
Programs, E, F and H had a system representation, but did not use it to transfer significant amounts of design information. Stakeholders for these programs interacted infrequently with the SR. Per the criteria, knowledge sharing was "weak" in these cases.

Program C's SR consisted of fielded prototypes that were in remote overseas locations. The prototypes were inaccessible to stakeholders involved in the design effort. Also, the design of the fielded prototypes differed substantially from the contractor's design for the new system. Therefore, no knowledge sharing came about as a result of interaction with the SR for this program.

Certain specific patterns of knowledge sharing mentioned by stakeholders played a clear role in adaptability. Interviewees indicated that SRs in six of the programs were used to identify issues and opportunities (potential collaborative changes). In four of these cases, the SRs also were used to perform "what if" analyses to evaluate potential changes.

Figure 6.5 captures the level of knowledge sharing using the SR and the adaptability level for each program. The figure also shows whether the programs used the SR to identify changes or to perform “what if” analyses to evaluate changes.

Knowledge Sharing with SR vs. Adaptability



DEFINITION OF CRITERIA (SUMMARY)
-Exceptional – in depth stakeholder interaction (fully exercising SR) on a daily basis
-Strong –substantive interaction (exercising some aspects of SR) on a daily basis
-Moderate – substantive interaction (exercising some aspects of SR) on a weekly basis or less
-Weak – top-level interaction (brief exposure to some aspects of SR) on a weekly basis or less
-None – no interaction

Figure 6.5. Knowledge sharing with SR versus adaptability

Six of the programs (A, G, D, F, E and C) demonstrate a trend in which higher levels of knowledge sharing, facilitated by a system representation, correlated with higher levels of program adaptability. Two programs, B and H, exhibited levels of adaptability

that were higher than other programs with comparable levels of knowledge sharing. Program B was able to elicit and disposition a large number of government comments on the software design during the course of a small number of interactive sessions with their SR. These changes, as described earlier, were low value and low risk, making it easier to process a large number of changes in a short period of time without dependence on frequent, iterative interaction with the SR. Program H encountered a number of changes in requirements and constraints after the start of the design phase, which forced several collaborative changes to be implemented. Many of these changes were identified and evaluated separately from the limited knowledge sharing provided by the Program H SR.

Six of the programs used the SR for identifying potential collaborative changes. The two exceptions were one “moderate” and one “low” adaptability program. Of these six, four also used the SR to evaluate changes. As will be discussed in chapter 7, identification and evaluation of potential changes are enablers of adaptability.

The information presented in Figure 6.5 provides the rationale for the findings listed below in Table 6.2.

Finding #1: Higher degrees of knowledge sharing using system representations corresponded to higher levels of program adaptability.
Finding #2: System representations were used as enablers for adaptability by assisting identification and evaluation of potential changes.

Table 6.2. Findings #1 and #2: System representations and knowledge sharing

Aspects of system representations that relate to their effectiveness in enhancing adaptability are discussed in the next section. Specific recommendations regarding stakeholder roles, which are also critical to adaptability, are explored in chapter 7.

6.4 Characteristics of Effective System Representations – SR Fidelity

This section addresses the research question: "What characteristics make SRs effective at promoting adaptability?" Interview data and program documentation was evaluated to identify SR characteristics that might influence the level of program adaptability. As a result, SR fidelity (i.e. realism in reflecting the system design) was investigated. SR fidelity manifested itself in two ways - level of detail and coverage.

Level of detail dealt with the degree to which the SR represented the entire design functionality. Coverage involved the degree to which the SR covered stakeholder emphasis areas, which were program priorities identified by user and SPO representatives during interviews.

To evaluate the first aspect of fidelity, system representations for the eight programs were characterized as providing one of three levels of detail regarding the system design functionality. Definitions for "system", "subsystem" and "minimal" levels are provided in Table 6.3.

Level of System Detail	Definition of Levels
System level	Portrays overall system functionality and interaction of subsystems
Sub-system level	Portrays subsystem functionality
Minimal	Minimal representation of functionality of the design

Table 6.3. Criterion for level of system represented

Programs A, B and G developed system representations that captured the full functionality intended by the system design, including the interaction of subsystems, as it was understood at the time of SR creation. Of the four most adaptable programs, only Program H had a SR with less than full "system" level detail. As described earlier, Program H responded to a series of changes in program requirements and constraints, which required collaborative changes. Characteristics of Program H's SR were therefore less relevant to the adaptability result than was the case for other programs.

In addition to H, programs D and E, which were moderately adaptable, also had a SR that provided "subsystem" level detail. Program D used a software spiral approach, evaluating multiple development versions of the software (the SR) before delivery. However, software segments were not integrated until just before test. Program E also used their development software as a SR. This SR reflected different portions of overall functionality at different times during SPO review sessions.

The system representations for programs C and F provided "minimal" design detail. For Program C, the SR consisted of overseas prototypes that were not accessible to stakeholders. Program F involved functional software development that was distributed among several contractors. Program F's SR was the prime contractor's

development software, and it reflected little system functionality until software integration took place. Over the course of the design phase, this SR provided minimal information about system functionality to government stakeholders.

Figure 6.6 below shows the trend of higher adaptability in cases of programs whose SR provided greater design detail. The information in this figure supports the finding listed in Table 6.4.

Finding #3: System representations that captured greater levels of design detail regarding intended system functionality were more effective in promoting adaptability.

Table 6.4. Finding #3: SR effectiveness and design detail

Detail Level of SR vs. Adaptability

Level of Representation / *Adaptability*	System Level Detail	Subsystem Level Detail	Minimal Design Detail
Very High	A, B		
High	G	H	
Moderate		D, E	C
Low			F

DEFINITION OF CRITERIA
- System level: portrays overall system functionality and interaction of subsystems
- Subsystem level: portrays subsystem functionality
- Minimal: minimal representation of functionality

Figure 6.6. Detail level of SR versus adaptability

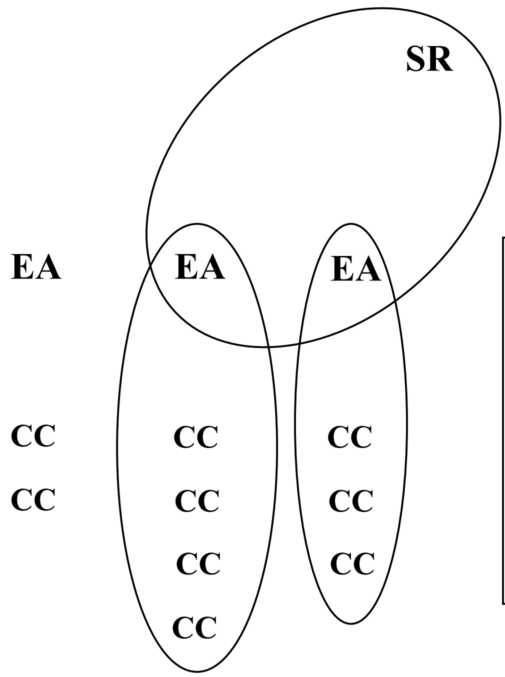
The second aspect of SR fidelity was coverage. SR coverage related to the concept of stakeholder emphasis areas (EAs), which were program priorities cited by SPO and user subject matter experts during interviews. Each program had one or more of these priority areas. Seven different EAs were identified during the case studies: technical performance (i.e. functionality), user interface, interoperability, maintenance, life cycle cost, reliability and development cost. All of the collaborative changes (CCs)

implemented by the programs were categorized as being in one of these same EAs. Also, the SR for each program was evaluated to see which of the program's EAs were reflected in the SR.

From these different elements, an indirect measure of SR coverage was derived. Figure 6.7 illustrates the measure of SR coverage, which was determined as follows:

- Establish the SPO and user emphasis areas (EAs) based on interview data.
- Evaluate which EAs were reflected in the SR based on interview data and program documentation.
- Determine the number of collaborative changes (CCs) in the SR-represented emphasis areas.

As an example, Program A had three emphasis areas: technical performance, maintenance and reliability. The SR for Program A covered all three of these emphasis areas, and 16 of the program's 19 collaborative changes were in these three EAs. Program F had two EAs – user interface and interoperability. The SR for this program covered the user interface but did not reflect interoperability. Five of Program F's collaborative changes were categorized as relating to the user interface EA.



EMPHASIS AREAS:

- *Technical performance*
- *User interface*
- *Interoperability*
- *Maintenance*
- *Life cycle cost*
- *Reliability*

Interpretation:
-Program has 3 emphasis areas (EAs)
-SR captures 2 stakeholder EAs
-Total of (4+3=) 7 CCs in the SR-reflected EAs

Figure 6.7. Example of SR coverage

The resulting numbers provided an indication of the level of SR coverage. The measure was indirect since it was not always possible to verify that the collaborative changes in the SR-represented areas were influenced by usage of the SR.

SR coverage for each program was determined to be either high or low, depending on the number of collaborative changes that were implemented in emphasis areas represented by the SR. The programs fell into two discrete groups based on this analysis. Programs A, B, E, and G were considered to have high coverage. These

programs had from 12 to 34 collaborative changes in SR-covered areas. The low coverage programs were C, D, F and H. They had between 2 and 5 collaborative changes in SR-covered areas. Also, high coverage programs had greater than 50% of their collaborative changes in SR-covered emphasis areas, and low coverage programs had 50% or less of their collaborative changes in SR-covered emphasis areas. Figure 6.8 presents this data, along with a matrix of coverage versus adaptability.

SR Coverage of Emphasis Areas (EAs) vs. Adaptability

High SR Coverage

Case	**CCs in SR EAs**	**CCs Total**
B	**34**	34
A	**16**	19
G	**12**	23
E	**12**	12

Low SR Coverage

Case	**CCs in SR EAs**	**CCs Total**
F	**5**	10
D	**3**	9
H	**2**	17
C	**2**	8

These collaborative changes (CCs) were in emphasis areas portrayed by the SR

	High SR Coverage	**Low SR Coverage**
Very High to High Adaptability	**A, B, G**	**H**
Moderate to Low Adaptability	**E**	**C, D, F**

OBSERVED GOVERNMENT EMPHASIS AREAS:

- *Technical performance*
- *User interface*
- *Interoperability*
- *Maintenance*
- *Life cycle cost*
- *Reliability*
- *Development cost*

Figure 6.8. SR Coverage of emphasis areas versus adaptability

Programs A, B and G, which were high coverage programs, achieved very high or high levels of adaptability. Programs C, D, and F had moderate or low adaptability and had low SR coverage. Programs E and H departed from the overall pattern of correlation between adaptability and coverage.

Program E had a SR that covered all four of its stakeholder emphasis areas, but the program was a low priority effort. The user had no interaction with the SR during the design phase. SPO engagement with the SR consisted of four informal sessions in which feedback was given to the contractor. The relatively light interaction of government stakeholders with the Program E SR may explain why this program had high SR coverage but only moderate adaptability. The SR was not used to its full potential.

The circumstances of Program H, involving a large number of user requirements and new constraints that were identified after the start of design, were discussed earlier. Many of the program changes concerned collaborative resolution of issues external to the design rather than exploration of the evolving design for improvement opportunities. Program H achieved a high adaptability through extensive discussions at user working groups and other meetings in which these external issues were resolved.

In summary, six of the eight programs fell into a pattern of correlation between coverage and adaptability, while the other two programs had circumstances that led to a departure from this pattern. This data led to the finding presented in Table 6.5.

Finding #4: System representations that covered stakeholder emphasis areas were more effective in promoting adaptability than system representations that did not.

Table 6.5. Finding #4: System representations and coverage

6.5 Discussion of System Representations and Analysis

The case studies showed that SRs were not always used to cover all seven of the identified stakeholder emphasis areas. The case study data that were collected and analyzed for system representations and emphasis areas are summarized in Figure 6.9. For each program, the figure identifies the stakeholder emphasis areas that were identified in the interviews and indicates whether the SR represented the area. The figure lists the number of collaborative changes in SR-represented emphasis areas and the total number of changes. It also identifies SRs that provided subsystem or minimal levels of detail.

In the cases where programs had emphasis areas that were not covered by their system representation, a "no" is indicated in the figure. The most notable example was interoperability, which was an emphasis area for five programs, but was only manifested in one of the program's SRs. Interoperability was problematic due to the necessity of simulating design details of other systems, which were typically out of the program's control. At the opposite extreme was the user interface, which was captured by a SR for all five of the cases in which it was an emphasis area. These observations raised the question of potential strengths and weaknesses of SRs, and also highlighted a potential role for analysis in support of knowledge sharing and decision-making.

Some programs commented on the effectiveness of using analysis to assist stakeholders in understanding implications of the design. Analysis is a very broad term, but it is typically thought of in terms of quantitative calculations rather than visual phenomena.

SR Coverage of Program Emphasis Areas (EAs)

Case	Stated Emphasis Areas (EAs) TP	UI	MX	RL	DC	LC	IO	CCs in SR EAs	CCs Total
B		SR						34	34
A	SR		SR	SR				16	19
G			SR	SR	SR	No		12	23
E	SR*	SR*	SR*				SR*	12	12
F		SR**					No	5	10
D		SR*					No	3	9
H	No	SR*		No			No	2	17
C	No		No	No	SR**		No	2	8

Key:
SR: SR shows EA
No: SR does not show EA
: Not an EA for this program
*SR limited to subsystem level detail
**SR showed minimal design detail

EMPHASIS AREAS:
TP: *technical performance*
UI: *user interface*
IO: *interoperability*
MX: *maintenance*
LC: *life cycle cost*
RL: *reliability*
DC: *development cost*

Figure 6.9. SR coverage of program emphasis areas

In some instances, analysis was used to fill a gap that could not be provided by a system representation. Program G, for example, had an emphasis on life cycle costs. The SR for the program did not reflect this emphasis area. The program developed and used a life cycle cost analysis tool to help make design decisions. Both the SPO and the

contractor indicated this cost model was an extremely valuable decision aid for potential program changes.

Since analysis was not the focus of this research, data on the subject was limited. However, some potential implications of the relative strengths and weaknesses of system representations and analysis are worthy of discussion. Both system representations and analysis have the capacity to help in knowledge sharing, and therefore in adaptability. These two mechanisms have different characteristics, which lead to different strengths and weaknesses.

System representations seemed to work best at portraying visual aspects of the design such as technical performance (defined for this research as functionality), the user interface, and maintenance. Interviewees that interacted with effective SRs were consistent in stressing the importance of being able to interact with the system in a hands-on manner as a means of understanding the functionality of the design and evaluating the implementation of the user interface. Stakeholders of the hardware programs that had access to a SR explored maintenance implications. One user described the effect of a SR through the saying, “a picture is worth a thousand words.” The only times that technical performance, user interface, and maintenance were emphasis areas and were not part of the SR were for Program C, which did not have access to their SR, and for Program H (in the area of technical performance), which was missing a government furnished radio during the design phase.

While data were not explicitly sought on analysis, some of the programs mentioned the importance of using analytical approaches. In particular, program interviewees indicated they did analysis in the areas of life cycle cost, development cost

and reliability. Program G stated that life cycle cost was a strong area for analysis. In the interviews, references to development cost and reliability implied that analysis of these areas was of moderate significance. Analysis was not an explicit focus of the research design, and data on this subject are insufficient to draw firm conclusions.

Two areas, reliability and development cost, were noted in the interviews as being addressed by analysis and by system representations. These areas illustrate the potential for system representations to be of help in non-visual areas. Running a SR such as a prototype or developmental software release allowed programs to gather data on the reliability of the design in its formative stages. In some programs, procuring elements of a system representation was mentioned as helping establish development cost figures. Based on these examples, SRs have the potential to reflect and transfer knowledge beyond visual aspects of the design. However, in such areas, analysis may also play a valuable role.

Figure 6.10 summarizes these observations, presenting a notional picture of the efficacy of system representations and analysis in representing details of the design for the seven emphasis areas. Given the limited amount of data on analysis, the placement of emphasis areas in the figure is not conclusive. However, the figure illustrates the potential for system representations and analysis to function as valuable complementary mechanisms for knowledge sharing about different aspects of the system design.

Efficacy of SR and Analysis for Different Emphasis Areas (Notional)

Efficacy of SR \ Efficacy of Analysis	Low	High
Low	Interoperability (IO)	Life Cycle Cost (LCC)
Medium		Reliability (RL) Development Cost (DC)
High	User Interface (UI) Technical Performance (TP) Maintenance (MX)	

Figure 6.10. Notional efficacy of SR and analysis for different emphasis areas

6.6 Summary of Findings and Theoretical Implications

This section addresses the connection between theoretical considerations for organizational adaptability that were developed in chapter 3 and the findings identified in this chapter related to system representations. The findings are consolidated in Table 6.6.

Findings
1. Higher degrees of knowledge sharing using system representations corresponded to higher levels of program adaptability. 2. System representations were used as enablers for adaptability by assisting identification and evaluation of potential changes. 3. System representations that captured greater levels of design detail regarding intended system functionality were more effective in promoting adaptability. 4. System representations that covered stakeholder emphasis areas were more effective in promoting adaptability than system representations that did not.

Table 6.6. List of findings for system representations

In chapter 3, concepts from the literature of complex adaptive systems (CAS) theory were explored and refined to derive organizational considerations for adaptability given an inter-organizational focus. The following three principles were derived:

- Look for and resolve potential perturbations to stability (underlying CAS principle - self-organization)

- Develop tools and procedures for information/knowledge sharing (underlying CAS principle - interaction)
- Balance structure and flexibility (underlying CAS principle - zone of novelty)

The correlation between findings and the CAS organizational constructs derived in chapter 3 is laid out in Table 6.7.

Correlation Between Findings and CAS Constructs

Research Questions:	**Finding:**	**CAS Organizational Construct:**	**CAS Principle:**
How does a SR enhance adaptability?	F1 – Higher knowledge sharing led to higher adaptability F2 – SRs assisted identification and evaluation of changes	Look for and resolve potential perturbations to stability	Self-organization
What characteristics make SRs effective?	F3 – Capture greater level of design detail F4 – Cover stakeholder emphasis areas	Tools, procedures for information sharing	Interaction
What are the roles of stakeholders?	See stakeholder roles in chapter 7	Balance structure and flexibility	Zone of novelty
Other program characteristics?	See program characteristics in appendix B	N/A - Alternate explanations	N/A – Alternate explanations

Table 6.7. Correlation between findings and CAS theory

Each of the rows in Table 6.7 traces a relationship from a research question and a finding to a CAS organizational construct with an underlying CAS principle. The following paragraphs present the case for correlation between two of the CAS organizational constructs and the two SR-related findings derived from the case study data. The other two rows of the table, relating to the third and fourth research questions, refer to matters addressed respectively in chapter 7 and appendix B.

<u>Look for and resolve potential perturbations to stability</u>

Self organization, or emergent order, is a key trait of complex adaptive systems. This research treats the three stakeholder organizations as a complex adaptive system. As described in chapter 3, for organizations to be adaptable in an inter-organizational setting, they must look for and resolve perturbations (i.e. program issues and opportunities). In the case of acquisition stakeholders, perturbations could involve such things as funding cuts, difficulties developing technology or user reprioritization based on learning from operation of current systems. The resolution process drives emergent changes in the program baseline. These emergent changes are ordered rather than chaotic ("emergent order") in the sense that the stakeholders consciously adopt them to remedy problems or to add value.

Stakeholders in the case studies shared knowledge about the system design (finding #1). System representations were a key enabler in this activity. More specifically, sharing of information with SRs facilitated identification and evaluation of potential changes (finding #2), which essentially amounts to looking for and resolving

perturbations. Therefore the findings (#1 and #2) about knowledge sharing are directly related to this CAS organizational construct.

Develop tools and procedures for information sharing

Tools and procedures that support information sharing are facilitating interaction between the stakeholders. Interaction is another key characteristic of complex adaptive systems, and is essential for organizational adaptability in an inter-organizational context.

For the programs that were studied, both system representations and analytical tools provided this facilitation. Findings 3 and 4 cover the value for programs of developing a SR that has a system level of detail and covers stakeholder emphasis areas. These two findings describe the characteristics of collaborative tools (in this case SRs) that make them more effective at supporting information sharing, establishing a tie between the findings and this CAS organizational construct.

Balance structure and flexibility

Chapter 3 included a discussion of the need for organizations to seek a balance between structure and flexibility so that their personnel will be in a "zone of novelty" in which sufficient control exists to prevent chaos but sufficient innovative effort is possible to promote adaptability. Chapter 7 aggregates data from the eight cases concerning the roles of the three primary acquisition stakeholders and provides case study examples that illustrate how performance of these roles contributes to the balance between structure and flexibility.

The correlation between these research findings and the CAS organizational constructs adds credence to the theoretical treatment of organizational adaptability in an inter-organizational setting that was developed in chapter 3. Discussion of the implications of these findings for CAS theory will be presented in chapter 8.

Chapter 7 Analysis of Stakeholder Roles

7.1 Introduction

Chapter 6 illustrated how system representations helped the SPO, the user and the contractor to share knowledge, leading to a common understanding of the evolving design that was an essential prerequisite for high levels of program adaptability. Achieving a shared understanding and reaching informed decisions on potential changes also required that stakeholders interact and evaluate shared information. Therefore, certain roles of the stakeholders, practiced in conjunction with usage of system representations, were found to be fundamental to effective adaptability. This chapter addresses the research question:

- **What are the roles of stakeholders in facilitating program adaptability?**

Adaptation took place in programs when a decision was made to implement a change in requirements or design. To reach such a decision, stakeholders needed to identify and evaluate potential changes. Changes that were approved had to be incorporated into the program baseline. Identification, evaluation and incorporation of changes can be thought of as adaptive functions performed collectively by the stakeholders. In highly adaptive programs, stakeholders were observed to perform roles that contributed in significant ways to these adaptive functions. This chapter consolidates the roles that were observed to be best practices supporting adaptability.

The chapter closes with an examination of the relevance of complex adaptive systems theory as a means of understanding effective stakeholder collaboration.

7.2 Adaptive Functions

Highly adaptive programs were found to share a common set of functions that contributed to program adaptability while helping to keep program risk to acceptable levels. These "adaptive functions" were the following:

- Demonstration of the partial design
- Identification of potential changes
- Evaluation of changes
- Incorporation of changes into the program baseline

In seven of the eight cases, programs used some form of system representation to demonstrate the partial design to government stakeholders. Programs that achieved higher levels of adaptability included efforts to ensure effective stakeholder participation in these demonstrations.

Stakeholders in most of the programs worked together to identify and evaluate potential changes. When done effectively, evaluation provided essential information to decision makers, and particularly to the SPO. This information included the anticipated risk and benefit of the change. After a decision, the SPO and the contractor worked together to incorporate changes into the program baseline.

Figure 7.1 shows the four adaptive functions taking place concurrently with execution of the design phase. The adaptive functions were iterative for most programs.

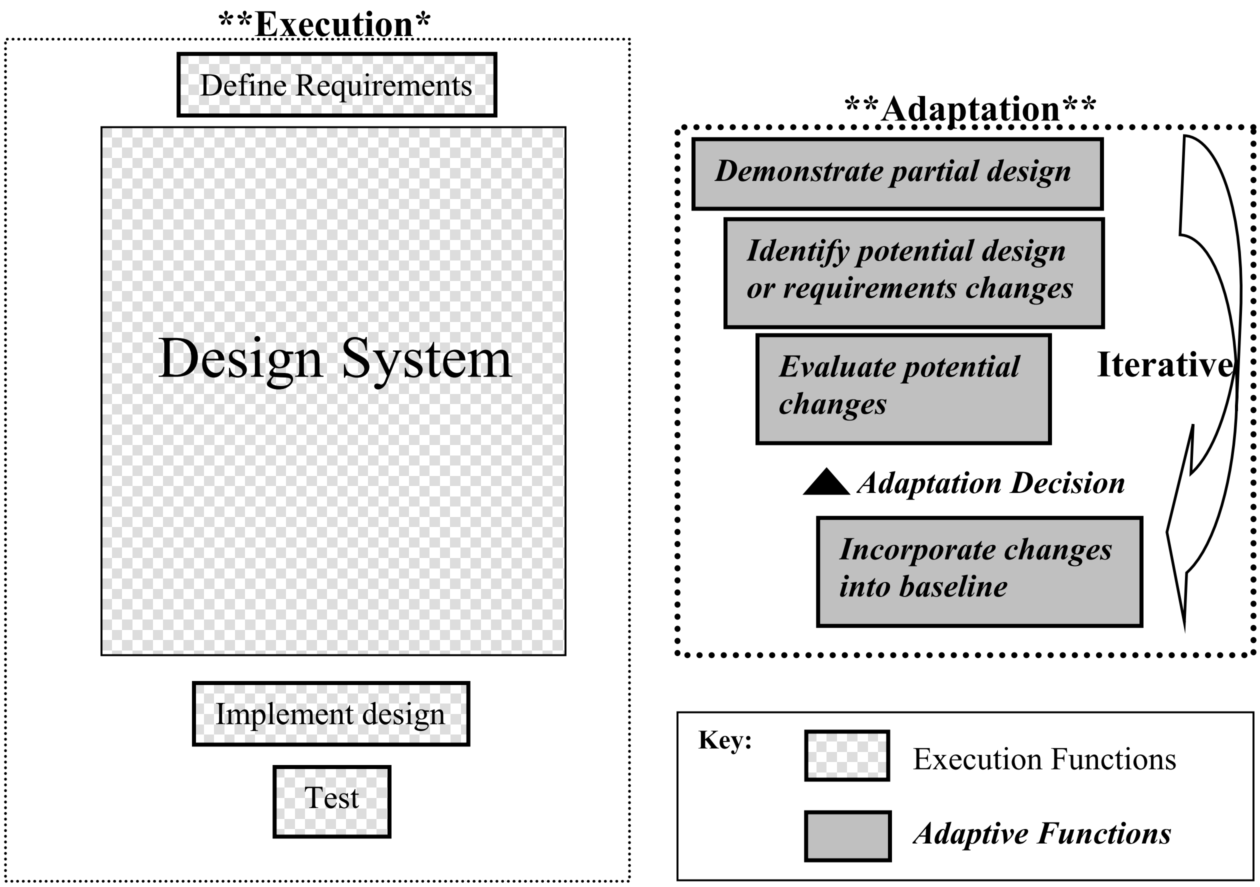


Figure 7.1. Adaptive functions during the design phase

Subsequent sections of this chapter describe the key SPO, user and contractor roles performed for each of these adaptive functions by the highly adaptive programs. Examples from the cases show high and low levels of stakeholder performance for each of the roles.

7.3 Stakeholder Roles

For most of the highly adaptive programs (A, B, and G), the stakeholder roles associated with the four adaptive functions acted in concert with usage of a system representation to facilitate adaptability. Program H provided an example of a program with strong stakeholder interaction but a limited system representation. Program E demonstrated that having an effective SR was not sufficient to produce high levels of adaptability when stakeholder collaboration was weak.

Programs A, B and G combined usage of effective system representations with strong collaborative interaction and effective evaluation of shared information. The best practices listed in the next three sections were observed in all three of these programs. Due to circumstances described in chapter 6, Program H achieved a high level of adaptability without having a strong system representation. Program H shared some but not all of the best practices observed in programs A, B and G. Programs C, D, E, and F lacked at least some of these best practices.

The stakeholder roles that were observed in these eight cases should not be construed to be a complete list of best practices for adaptability. However, their presence in the most adaptive programs, as well as their absence in less adaptive programs, provides evidence of their significance.

7.3.1 SPO Roles

Table 7.1 lists the key SPO roles that were identified for each of the four program adaptation functions. The table also provides examples taken from the eight cases of high performance and low performance in these roles. Roles in bold with gray backgrounds are best practices supported by case study data. Non-bolded roles are either common practice that are well understood or are only partially supported by research data.

The paragraphs following Table 7.1 describe these SPO roles based on observations from the case studies. The bolded entries from the table were important ingredients in creating an adaptive environment of collaboration among stakeholders. A brief discussion of the non-bolded areas is provided to explain why they were considered less significant or less conclusive.

SPO Roles for Adaptability

Function	SPO Role	High Performance (Examples from cases)	Low Performance (Examples from cases)
Demonstrate partial design	**Encourage and facilitate user engagement during the design phase**	Emphasized importance of user involvement from the start of design; user shown that inputs made a difference	Discouraged user from participation during design phase (SPO felt user role was limited to defining requirements)
	Manage user expectations	Briefed user on current and future SR capabilities in preparation for user interaction with SR	Decided not to share SR with users until substantial functionality was available due to concern that user would criticize program
Identify potential design or requirements changes	Provide design feedback: system considerations and "ilities" (reliability, maintainability, interoperability…)	Tracked system's ability to flow data to meet all user needs; Analyzed technical risk areas (e.g. antenna interference and COTS performance in operating environment) to ensure system reliability	Allowed contractor to develop and demonstrate system in non-integrated segments; Lacked process for tracking related systems that were in development to spot future interoperability issues
Evaluate potential changes	**Facilitate contractor evaluation**	Established contract provisions for studies; Encouraged contractor "what if" exercises	No resources planned for contractor "what if" exercises
	Evaluate risks	Assessed realism of cost estimates; Weighed added risk to meeting constraints	Underestimated resources required to implement changes
Incorporate changes into baseline	Issue rapid approval	Added work scope and funds to contract quickly	May have delayed timely implementation due to under staffing (not conclusive)

Notes: Roles in **bold with gray backgrounds** are best practices supported by case study data
Non-bolded roles are either common practice (well understood) or are only partially supported by data
Examples are from case study data involving specific practices that influenced program adaptability

Table 7.1. SPO roles for adaptability

~**Encourage and facilitate user engagement during the design phase**

Without exception, the most adaptive programs (A, B, G and H) had heavy user involvement during the design phase. However, this involvement did not come about without considerable encouragement and planning from the SPO. Users in all programs described themselves as busy, and frequently indicated that they struggled to define what role, if any, they should play in design. It was therefore essential that the SPO continually be proactive to encourage user engagement and provide evidence that user involvement was adding value for the new system.

SPOs that were successful in this role focused on early contact and frequent dialog with the users to encourage participation during the design phase. They ensured sessions with the user were relevant and useful and that user feedback was given serious consideration. The strongest examples of valuable sessions were ones in which the users were able to interact with the system representation and provide feedback. Users consistently voiced enthusiasm and acknowledged the importance of these sessions.

By contrast, users that felt unwelcome or unproductive when engaging during the design phase tended to minimize their involvement in favor of higher priority activities.

~**Manage user expectations**

Once users started to be engaged in evaluating the design, it became critical for the SPO to manage expectations about the maturity of the design at the point in time of user exposure. SPO SME's reported cases in which they failed to manage expectations and indicated the result was severe damage to the credibility of the SPO and the contractor.

User representatives were typically interested in ensuring the new system would have all of the required functionality, and field users in particular were often unaware of the practice and implications of sharing a design with partial functionality to solicit early feedback. SPO SME's indicated that if user representatives were briefed on the specific functionality inherent in the system at the time of interaction and made to understand the intent of such an incremental review process, they were able to concentrate on the functionality that was present rather than focusing on missing features that were not yet incorporated in the design.

~Provide design feedback: system considerations and "ilities" (reliability, maintainability, interoperability…)

This observation was potentially significant, but the role of the SPO in providing design feedback varied widely between different programs. The technical context of the program and the background of SPO personnel played a strong role in the nature of SPO contributions to the design. Therefore, it was not possible to structure a specific recommendation as to the role of the SPO in providing design feedback.

The most valuable SPO contributions to contractor design observed in the eight cases involved system considerations (particularly in case H) such as reliability, maintainability and interoperability.

~**Facilitate contractor evaluation**

The program A, B, and G SPOs (as well as some others) exhibited up-front awareness of the likelihood, and in some cases the desirability, of change. These SPOs anticipated the need for the contractor to evaluate potential changes. The Program A

SPO included a study clause in the contract and actively encouraged “what if” analysis of potential value-adding changes. Programs A, B and G designated resources for such analyses in advance. Less adaptive programs frequently had no resources identified for the contractor to perform “what if” efforts, and potential changes were typically viewed by the SPO as problems rather than as opportunities.

Interviews with the SPOs for programs A, B and G showed a common enthusiasm for potential changes and a proactive encouragement of contractor innovation. The Program H SPO was under budgetary pressure, but they also encouraged their contractor to evaluate potential contract changes to ensure the consequences of adaptation decisions would be understood.

~**<u>Evaluate risks</u>**

It was essential in programs that were considering changes to requirements or design for SPOs to evaluate risks that cost and schedule constraints might be violated.

> The Program A SPO indicated that it was necessary to discourage users from seeking changes that involved too much technical risk.

Program G made a decision late in the design phase to adopt a less advanced computer option to mitigate program risk. The SPO SME indicated this decision was a key to the program’s success in meeting cost and schedule baselines.

Program H was forced by schedule pressure to implement a contract change before the cost implications were clearly understood. The SPO SME reported that these changes became the source of considerable friction with the contractor, and added more financial and schedule pressure to the program.

The consistent trend in highly adaptable programs was that the SPO acted as a gate to determine whether contemplated changes would add unacceptable levels of program risk. Contractor input (see contractor roles section) on implementation approach, cost and benefit of changes was an essential enabler for this SPO role.

~Issue rapid approval

SPOs that processed changes quickly and efficiently helped the contractor implement the changes with minimal disruption to program execution. This observation was considered to be a standard practice, although some programs mentioned delays perceived to be associated with SPO under-staffing. The under-staffing rationale was not considered conclusive because it may have been possible to speed up approvals through more efficient change management processes.

7.3.2 User Roles

Table 7.2 provides the list of key user roles observed for the four functions. These roles are discussed below.

~**Coordinate field participation**

User headquarters personnel for most of the programs took an active role in coordinating opportunities for experienced field personnel to get exposure to the evolving design. In these cases, field users were able to provide feedback on the design, as

described below. Field personnel for programs C, E and F were exposed to minimal or no design information during the design phase.

~**<u>Provide design feedback: operational perspective (how system will be used)</u>**

Field users with knowledge of current operational systems provided design feedback on how they felt the new system would be used. This input was cited by contractor SME's as being of tremendous value in making the design more suitable and effective for the user. User feedback constituted a means of achieving incremental validation that the design selected by the contractor was meeting user needs. Contractors for less adaptive programs noted that they often had to make design decisions in the absence of user input, speculating on how the system would be used in the field.

User Roles for Adaptability

Function	User Roles	High Performance (Examples from cases)	Low Performance (Examples from cases)
Demonstrate partial design	**Coordinate field participation**	Designated user headquarters coordinated involvement of future field users who had experience operating existing systems	Did not participate in review of contractor's design, or had review of design by user headquarters personnel only
Identify potential design or requirements changes	**Provide design feedback: operational perspective (how system will be used)**	Commented on how operators would use the system - led to design improvements and changes in requirements	Had minimal or no user interaction after initial requirements definition
Evaluate potential changes	**Define priorities (importance of potential changes)**	Updated priority list weekly; leadership emphasized importance of establishing and communicating clear priorities	Had minimal or no user interaction after initial requirements definition
Incorporate changes into baseline	**N/A**	**N/A**	**N/A**

Notes: Roles in **bold with gray backgrounds** are best practices supported by case study data
Non-bolded roles are either common practice (well understood) or are only partially supported by data
Examples are from case study data involving specific practices that influenced program adaptability

Table 7.2. User roles for adaptability

~**Define priorities (importance of potential changes)**

All of the adaptive programs (A, B, G, and H) had a mechanism for establishing user priorities. User A senior management put an emphasis on defining and communicating priorities for the program on a continuous basis. Contractor G used a weekly list of issues and tasks to promote discussion of priorities and align user expectations. Regardless of the mechanism that was employed, user input on the priority

of potential changes and on options to delay existing requirements to make changes affordable was essential to proper change evaluation.

Program E provided an example in which the user was not engaged during design, in part due to the low priority of the effort. Contractor E adopted a practice of consulting with test personnel who were in-plant for another effort to determine the relative merits of potential changes, but the lack of a coherent picture of user priorities may have reduced the opportunities to add value to the product during design.

7.3.3 Contractor Roles

The key contractor roles are contained in Table 7.3 and discussed below.

~ **Create and share SR**

Contractor considerations related to system representations were discussed in depth in chapter 6. These considerations included the need to facilitate knowledge sharing with the SR and the importance of SR fidelity (level of design detail and coverage of stakeholder emphasis areas.)

~ Select design options to meet requirements

The role of selecting design options to meet requirements is the contractor's central responsibility during the design phase, so it was considered a standard practice.

Contractor Roles for Adaptability

Function	Contractor Roles	High Performance (Examples from cases)	Low Performance (Examples from cases)
Demonstrate partial design	**Create and share SR**	See SR discussion and findings in Chapter 6 regarding knowledge sharing and SR fidelity	See SR discussion and findings in Chapter 6 regarding knowledge sharing and SR fidelity
Identify potential design or requirements changes	Select design options to meet requirements	Standard part of work effort – no appreciable differentiation between programs	Standard part of work effort – no appreciable differentiation between programs
Evaluate potential changes	**Evaluate cost, benefit and best implementation approach**	Assessed the benefit of changes and the work effort required for implementation; explored implementation options	Responded to user requests with minimal consideration of cost and schedule impacts
Incorporate changes into baseline	Update SR	Incorporated changes and provided iterative opportunities for SR review	Made limited (or no) iterations of SR available for government review
	Update program documentation	Ensured thorough documentation of all changes	Captured agreements inconsistently

Notes: Roles in **bold with gray backgrounds** are best practices supported by case study data
Non-bolded roles are either common practice (well understood) or are only partially supported by data
Examples are from case study data involving specific practices that influenced program adaptability

Table 7.3. Contractor roles for adaptability

~ **Evaluate cost, benefit and best implementation approach**

In order to understand the risk associated with potential changes, the contractor had to develop an understanding of the probable implementation approach and the cost and schedule implications of changes. The contractor for programs A, D, G and H used their system representation to try out different approaches and gather information on the

cost and benefit of changes. Most program B changes were considered low cost and low risk, so the primary change evaluation for this program consisted of determining the benefit of the change to the warfighter.

~Update SR

Highly adaptable programs incorporated government feedback in the SR and provided additional opportunities to generate further feedback. This role supported an iterative approach to government evaluation of the design. Updating the SR to reflect changes was considered a standard practice for programs.

~Update program documentation

Efficient contractors stayed on top of program documentation to ensure agreements on program changes were not forgotten. This role was also considered to be standard practice.

7.3.4 Summary of Roles

Table 7.4 lists the program adaptability functions and the associated key stakeholder roles that were identified earlier.

Function	SPO Role	User Role	Contractor Role
Demonstrate partial design	• Encourage and facilitate user engagement • Manage user expectations	• Coordinate field participation	• Create and share SR
Identify potential design or requirements changes		• Provide design feedback: operational perspective (how system will be used)	
Evaluate potential changes	• Facilitate contractor evaluation • Evaluate risks	• Define priorities (importance of potential changes)	• Evaluate cost, benefit and best implementation approach

Table 7.4. Key stakeholder roles supporting program adaptability functions

The next section applies theoretical considerations and observations from the cases to discern patterns of stakeholder roles and explore the significance of these patterns.

7.4 Theoretical Implications

Theorists studying complex adaptive systems have described the concept of a zone of novelty in which adaptation is unleashed in a system but sufficient control is maintained to avoid chaos (Waldrop, 1992; Kauffman, 1995; Holland, 1995). The actions of the agents in a complex adaptive system create a balance between order and chaos, poising the system in this novel zone. Analysis of the case study data uncovered

patterns of behavior for the SPO, the user and the contractor that created just such a zone of novelty in the Air Force acquisition context. Some stakeholder behavior patterns, or roles, encouraged adaptation by promoting flexibility, while others afforded sufficient structure to prevent such chaotic aspects of change as cost growth and schedule delays.

Figure 7.2 divides the previously defined stakeholder roles that influenced adaptability into those that provided "structure" and those that contributed to "flexibility."

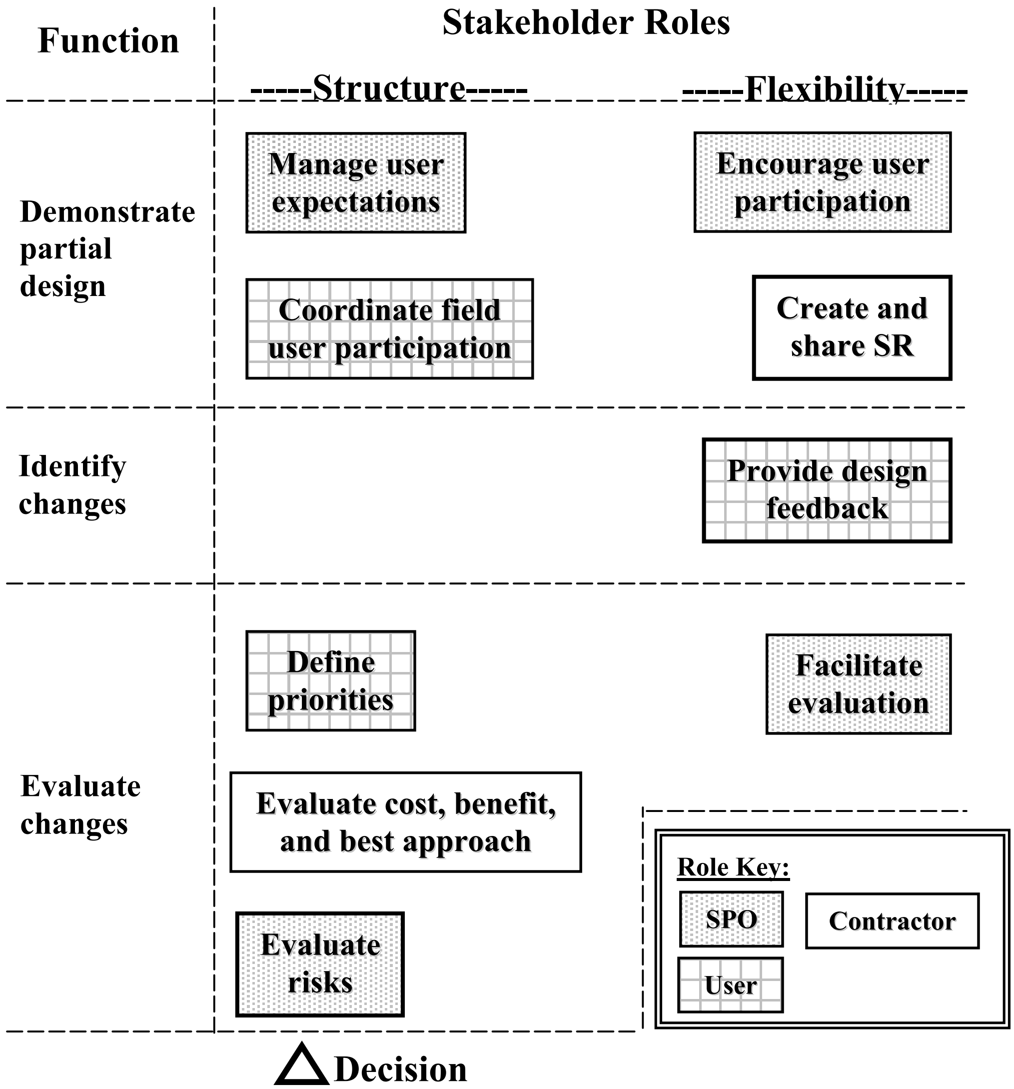


Figure 7.2. Structure and flexibility from stakeholder roles

Taken together, the roles listed in Figure 7.2 created a zone of novelty for the stakeholders associated with programs A, B and G in which they were able to excel at adapting program requirements and design. Table 7.5 describes the impact to adaptability if roles are not filled. As the table indicates, each of these stakeholder roles

is essential to one or more of the adaptive functions needed to achieve a high level of program adaptability while managing risk to program constraints.

Stakeholder Role	Impact if Role Not Filled
Manage user expectations	User frustration due to missing functionality leads to breakdown in stakeholder collaboration
Encourage user participation	Lack of user participation prevents identification of potential changes
Coordinate field user participation	Reviewers without knowledge of intended system usage cannot provide meaningful feedback on the design
Create and share SR	Lack of shared understanding of the design inhibits identification of potential changes
Provide design feedback	Potential changes not identified
Define priorities	Evaluation of potential changes is ineffective due to lack of information on value of changes to user
Facilitate evaluation	Contractor lacks resources or permission to evaluate potential changes
Evaluate cost, benefit and best approach	SPO has insufficient information to evaluate potential changes
Evaluate risks	Program runs the risk of destabilizing cost and schedule growth due to implementation of changes

Table 7.5. Value and impact of stakeholder roles

Collectively, the adaptability functions and associated stakeholder roles served the purpose of providing essential information to decision-makers who were responsible to disposition potential changes. Decision options observed in case studies included implementing the change, disapproving it, modifying it to mitigate risk, delaying it to a

future delivery, or implementing it in conjunction with other program changes to manage overall program risk.

Another manifestation of the concept of a "zone of novelty", as introduced in section 3.6.2, is Highsmith's (2000) concept of "time boxing." Managing using time considerations provides a way for programs to encourage creativity within temporal constraints such that key trades are explored and resolved without delaying program execution. In the eight cases, several examples of time considerations were observed. The most significant examples are summarized in Table 7.6.

Case	**Description of Time Consideration**
A	Users operating the system provided timely feedback, allowing shaping, evaluation and implementation of changes within the time constraints of incremental deliveries.
B	The contractor felt feedback on the design would have been useful earlier.
B	Using the SR during design provided a means to get diverse users together to make government decisions – led to valuable inputs that shaped the design.
D	Contractor committed to respond to issues and actions in one week.
F	Management decisions were hampered by diverse priorities between two government SPOs. Even simple decisions were described as slow and painful. The program struggled until government decision processes were restructured.
G	A technology insertion option was structured early, held open until more information was available, and declined late in the program to avoid an unacceptable level of risk. The program was able to make an informed design decision.
H	The contractor lacked user input and had to make some design choices internally to keep the program on track.

Table 7.6. Time considerations from cases

The primary theme that emerged from these observations was that contractors wanted timely input from the government so they could make informed design choices. All of the programs were conscious of time considerations. When contractors lacked timely design feedback from the government, they made design decisions internally to keep the program schedule on track. The most adaptive programs found creative ways to share design information early, get timely feedback, and perform evaluation and implementation of changes within the design phase timeline.

The conclusion to be drawn from this data is that timeliness is a factor in the effectiveness of the following stakeholder roles: create and share SR; provide design feedback; define priorities; evaluate cost, benefit and best approach; and evaluate risks. These roles enable informed decisions on potential changes. It is important to note that data collected in this research focused on collaboration rather than internal contractor processes, so non-collaborative contractor activities may have involved additional time considerations that were not brought out in the interviews.

7.5 Finding on Stakeholder Roles

The finding associated with stakeholder roles is based on the previously discussed roles and their significance as discussed in section 7.4. The finding is presented in Table 7.7.

Finding #5: Performance of the following stakeholder roles gave programs the flexibility and structure needed to adapt while actively managing program risk. • ***SPO roles: encourage user participation; manage user expectations; and evaluate risks associated with implementing potential changes*** • ***User roles: provide design feedback on how the system will be used; coordinate field user participation; and define priorities*** • ***Contractor roles: create and share the system representation; and evaluate cost, benefit and best approach for potential changes***

Table 7.7. Finding #5: stakeholder roles

For the system of stakeholders and stakeholder interactions that is the subject of this research, the enabler for achieving a balance between order and chaos is performance of the identified stakeholder roles. As they fill these roles, stakeholders poise the system between a condition of excessive order in which opportunities to add value are missed and a chaotic condition in which the implications of changes are not well understood.

Chapter 8 provides a summary of the research findings as well as discussion of the implications of the research.

Chapter 8 Summary and Discussion

8.1 Introduction

This research has investigated the collaborative behavior of stakeholders and the adaptive results achieved during eight U.S. Air Force acquisition programs. Each program was studied during its design phase. Particular attention was given to the usage and impact of "system representations" – mechanisms such as prototypes or beta software releases that provided partial design information to stakeholders as the design evolved. Questionnaires, interviews and extensive reviews of program documentation were used to capture the patterns of interaction between the lead Air Force acquisition agency, or System Program Office (SPO), the contractor tasked with developing the new system, and the Air Force operational user community. The primary objective of the research has been to gain insight into why some programs are more adaptable than others in responding to challenges and opportunities present during the design phase.

Analysis of the data from the eight case studies has led to a set of findings that address the following research questions:

1. How does a system representation enhance adaptability?
2. What characteristics make system representations effective at promoting adaptability?
3. What are the roles of stakeholders in facilitating program adaptability?
4. Do certain characteristics of programs (requirements uncertainty, funding level and duration of design phase) predispose them to be more or less adaptable?

The analysis in Chapter 6 produced findings that addressed questions 1 and 2, related to system representations. Chapter 7 addressed stakeholder roles and provided a finding that responded to question 3. Analysis revealed that the program characteristics referred to in question 4 had no correlation to the level of program adaptability (see Appendix B). The complete set of findings is provided in the next section of this chapter.

The chapter continues with discussion of specific recommendations to policy makers and practitioners. Then, observations are presented regarding the implications for theory that result from this research. Suggestions are provided for further research. The chapter finishes with closing thoughts on the contribution represented by this work.

8.2 Summary of Findings

Table 8.1 lists the findings according to the research question addressed.

Findings -- System Representations:

Question 1: How does a system representation enhance adaptability?

- **Higher degrees of knowledge sharing using system representations corresponded to higher levels of program adaptability. (Note: this finding provides evidence that SRs enhanced adaptability because they facilitated knowledge sharing.)**
- **System representations were used as enablers for adaptability by assisting identification and evaluation of potential changes.**

Question 2: What characteristics make system representations effective at promoting adaptability?

- **System representations that captured greater levels of design detail regarding intended system functionality were more effective in promoting adaptability.**
- **System representations that covered stakeholder emphasis areas were more effective in promoting adaptability than system representations that did not.**

Findings -- Stakeholder Roles:

Question 3: What are the roles of stakeholders in facilitating program adaptability?

Performance of the following stakeholder roles gave programs the flexibility and structure needed to adapt while actively managing program risk.

- **SPO roles: encourage user participation; manage user expectations; and evaluate risks associated with implementing potential changes**
- **User roles: provide design feedback on how the system will be used; coordinate field user participation; and define priorities**
- **Contractor roles: create and share the system representation; and evaluate cost, benefit and best approach for potential changes**

Finding – Program Characteristics:

Question 4: Do certain characteristics of programs predispose them to be more or less adaptable?

Requirements uncertainty, funding and design phase duration did not have a causal relationship with respect to levels of program adaptability.

Table 8.1. List of findings

8.3 Acquisition Policy Recommendations

8.3.1 Acquisition Planning and System Representations

The correlation established in Chapter 6 between system representation (SR) usage and adaptive results makes a strong case that it would be valuable for many programs to use a SR during the design phase of program development. However, programs have resource limitations that could impact their ability to generate and modify a SR. The level of fidelity of the SR also raises resource considerations. Each program must balance their need for adaptability with their resource constraints.

The presence of the following factors, among others, can influence a program's need to adapt: changing technology, a changing threat, evolving or emergent user needs, and a rapid learning curve from operation of fielded systems. Given the presence of these factors, or others causing user or SPO knowledge pertaining to the new system to change over the course of the design phase, the value of a SR during design can be significant. For programs facing few of these uncertainty factors, expending resources for a high fidelity system representation, or for any level of SR, may be unnecessary.

Given these considerations, Air Force acquisition policy should be augmented to encourage acquisition planning prior to contract award to address evaluation of the potential benefit of a SR and consideration of required resources. As contract terms are structured in an Air Force request for proposal (RFP), allowances should be made for contractors to propose resources for creation, sharing and modification of a system representation. Also, the contractor should be encouraged to plan resources to evaluate and implement potential changes. This up-front planning will greatly facilitate the ability of the program to adapt as challenges and opportunities are identified.

8.3.2 User Involvement in the Design Phase

One of the challenges faced by the acquisition community is the fact that the user community is not familiar with the acquisition process. This unfamiliarity is particularly pronounced regarding contractor activities during the design phase, when user requirements are transformed into contractor-selected applications of technology. Yet since acquirers and contractors in turn lack full comprehension of the user's perspective (and in particular the details of how the system is likely to be used), there is a significant role for the user to play in the design phase of acquisition. The user role is well established for requirements generation and for testing of new systems. However, one striking insight of this research was that, in the design phase, many users were uncertain of the role they should play.

Users frequently indicated they were not able to envision all the implications for use of the new system when they were writing requirements and concepts of operation. Many programs did not have a concept of operation in written form. In several of the case studies in which users were able to visualize and interact with a partial design through a system representation, their knowledge of how the system was likely to be used enabled them to provide valuable input to the contractor regarding the evolving design. This mode of operating gave user representatives both a role in design and a mechanism to be productively engaged in the design process.

None of the eight cases studied involved a situation in which the contractor or SPO had the same grasp of operational considerations as was present in the user community. In almost all cases, the contractor specified either that user input had been extremely valuable to guide system design or that lack of user input had forced design

choices in the absence of knowledge of how the system would be used. Contractors in some cases made both comments about the same program.

These observations imply that the user community should be actively encouraged to participate in the design phase of weapon systems acquisition. Usage of a system representation during design greatly facilitates establishment and continuation of this interaction, since users are able to perceive the value of their participation and are more likely to continue engagement in the face of conflicting priorities.

During the design phase, the following user roles should be established in DoD and Air Force acquisition policy: a responsible user headquarters must coordinate participation of knowledgeable field representatives; these representatives must provide design feedback on how the system is likely to be used; and the user community must provide integrated, up to date priorities for the program as the design evolves.

8.4 Recommendations for Practitioners

8.4.1 Creating Effective System Representations

Several practical considerations related to creation of effective system representations emerged from the eight case studies. The SPO and contractor should consider the following aspects of SR creation during design:

- System representations should be created and shared at a system level of detail early enough to allow feedback from government stakeholders to influence the design.

- If programmatic constraints make implementation of a system-level SR impractical, a sub-system level SR should be created and shared early enough to allow feedback from government stakeholders to influence the design.
- Government stakeholders should explicitly define their priorities, or emphasis areas, for the program, and encourage the contractor to include representation of these emphasis areas in the SR to the extent practical. Emphasis areas described by stakeholders in the case studies included: user interface, interoperability, technical performance (i.e. functionality), maintenance, reliability, development cost and life cycle cost. As discussed below, some emphasis areas lend themselves to capture by a SR more readily than others.
- The following emphasis areas are visual in nature: technical performance, user interface, and maintenance. SRs in the observed cases were strongest at portraying these emphasis areas.
- Reliability, development cost and life cycle cost are emphasis areas that lend themselves to computational assessments. These areas are likely to be more approachable by analysis than by usage of a SR.
- Analysis and system representations can be used effectively as complements by the same program to cover a wide range of stakeholder emphasis areas.

8.4.2 Using System Representations

Analysis of the eight cases also yielded a set of practices that were important for effective usage of system representations. These practices were:

- Once the contractor can create a SR with enough detail to make aspects of functionality visible, there may be value in sharing the system representation with government stakeholders. Early usage of a SR has the advantage of soliciting early feedback, but it is essential to manage the expectations of operational users (as described in section 8.4.3) when partial functionality will be presented.
- Effective system representations made aspects of the design visible and provided the opportunity for stakeholders to interact with the SR.
- More in-depth and frequent stakeholder interaction with a SR tended to achieve greater knowledge sharing. This consideration must be balanced against resource limits and the need for contractor personnel to be able to execute design work in accordance with established timelines.
- Stakeholders should be encouraged to use the program's system representation to identify and evaluate potential changes during the design phase.
- SR's may offer insight into non-visual areas such as development cost and reliability in which creating and/or using the SR provides additional data about the design.

8.4.3 Creating a "Zone of Novelty"

The concept of a "zone of novelty", in which adaptability and program execution are both possible, was discussed at a theoretical level in chapter 3 and analyzed in depth in the context of stakeholder roles in chapter 7. The conclusion reached in chapter 7 was that there is a need for both flexibility and structure during the design process to establish this zone for the three primary acquisition stakeholders (SPO, user and contractor).

Flexibility is important to encourage creativity and innovation, while structure is essential to ensure proper awareness and consideration of cost and schedule constraints. Both aspects were present in the most adaptable programs, permitting stakeholders to function in a "zone of novelty."

Flexibility and structure were provided through the actions of stakeholders as they took on specific roles. The roles needed to create a "zone of novelty", as identified through analysis of the case studies, are summarized below:

- The following central roles should be performed to facilitate adaptability:
    - Contractor: create and share a system representation
    - User: provide design feedback on how the system will be used (based on an operational perspective)
    - SPO: evaluate risks associated with implementing potential changes

- The following supporting roles should be performed to facilitate the central roles:
    - SPO: encourage user participation and manage user expectations
    - User: coordinate field user participation and define priorities
    - Contractor: evaluate cost, benefit and best approach for potential changes

Two sets of roles were identified during analysis that must be executed in balance with each other:

- As SPOs encourage user participation, they must also be careful to manage user expectations. Specifically, it is necessary to inform user representatives of the level

of design maturity at the time of user evaluation and of the intent to provide partial design information to solicit incremental feedback. This notification of content and context (Highsmith, 2000) gives user representatives an appropriate perspective as they engage in evaluation of the design. Failure to manage expectations in some cases led to distrust and a breakdown in stakeholder collaboration.

- As users provide design feedback, identifying potential changes, the SPO must evaluate the risks associated with implementation of these changes. By performing this risk management function, the SPO is able to guard against the possibility that efforts to adapt might lead the program into an unstable condition from a cost and schedule standpoint. One method for managing these risks is to consider a range of options for each potential change, including the following:
    - Approve the change.
    - Disapprove the change.
    - Approve the change with modification(s) to mitigate risk.
    - Delay the change to a future incremental delivery if the change posed a risk to execution of the current delivery.
    - Implement the change in conjunction with other change(s) (potentially including deletion of other requirements) to manage overall program risk.

Time considerations (Highsmith, 2000) play a role in achieving a balance between adaptability and progress toward design completion. For this balance to be achieved, certain stakeholder roles that enable informed decisions on potential changes must be performed within the window of design activities. These roles are: create and share a SR; provide design feedback; define priorities; evaluate cost, benefit and best approach;

and evaluate risks. Programs should make use of contractor insight into timing of the design process to ensure these roles can be fulfilled while design trade spaces are still open.

One final observation on stakeholder roles involves the current culture of the SPOs, users and contractors. These stakeholders in the acquisition process share a strong focus on the importance of processes that has its origins in both engineering and management training. Effective collaboration requires an emphasis on relationships as well as processes. The significance of fostering and maintaining relationships is not ingrained in the culture of these organizations, which implies that full implementation of the collaborative roles described above will require widespread cultural changes. In 2003, the Air Force instituted a mandatory "Discovery Map" training for all acquisition personnel to emphasize the importance of collaboration as a new cultural paradigm. However, effective implementation of this cultural transformation will require more detailed, focused training that includes specific examples to illustrate the techniques and benefits of collaborative interaction between stakeholders. This training should ideally be given to program teams, including all major stakeholders whose success at collaboration will be a factor in achievement of program success.

8.5 Implications for Theory

Chapter 3 explored the implications of systems theory and complex adaptive systems (CAS) theory to stakeholder collaboration and adaptability. One of the compelling observations to come out of taking a systems view is the importance of understanding interaction of system components, such as the primary stakeholders in the

acquisition context, in order to capture overall system characteristics. CAS theory adds the insight that interaction of agents is a key element in an ongoing process of self-organization, or adaptation. In the context of this research, stakeholder collaboration is the engine for adaptability and knowledge sharing to build a common understanding between stakeholders is the central element of stakeholder interaction.

The application of CAS theory to organizations in chapter 3 led to the formulation of guiding constructs that may influence the capacity of organizations to be adaptable. This theoretical base was combined with the concept of a type of boundary object (Star, 1989; Carlile, 1997, 2002) called a system representation (SR) that provides a means to share information across organizational boundaries. Taken together, CAS theory and the concept of system representations provided the basis for construction of the four research questions explored during this study.

Chapters 6 and 7 include sections on the significance of CAS theory in shedding light on stakeholder interaction and resulting levels of program adaptability. The organizational CAS constructs listed below were evaluated in these two analysis chapters.

- Look for and resolve potential perturbations to stability
- Balance between structure and flexibility
- Develop tools and processes for information sharing

The CAS constructs correlated well with key findings of the research (see Table 6.7 and section 7.4), implying that CAS theory may be meaningful to apply in other inter-organizational settings in which the sharing of information to establish a common understanding has a relationship with adaptability. CAS theory projects that interaction

of agents will lead to self-organization, a phenomena that was observed in each of the eight cases in the form of program adaptations. The correlation between these research findings and the CAS organizational constructs adds credence to the theoretical treatment of organizational adaptability in an inter-organizational setting that was developed in chapter 3.

System representations seemed to be effective boundary objects in the sense that they helped stakeholders to bridge knowledge boundaries. In the context of system design, SR's were valuable because they presented the same visual information about the design to all stakeholders. A pattern has developed across many bodies of research (Robertson, 1990; Carlile, 1997 and 2002, Bernstein, 2000) that indicates visual representations lead to a shared understanding of information and facilitate collaborative evaluation and decision making during the design process.

In one of the most striking examples, Robertson (1990) studied how Computer Aided Design (CAD) facilitated the "transfer of design information within and between groups and companies in the product development process." He found that CAD systems "have an important role in the communication of design information" and that there was "a direct link between communication activity and performance" of design engineers who had to "coordinate design activities, resolve design problems, get new ideas for possible design alternatives, stay abreast of technological developments, or develop synergistic relationships with others." Robertson concludes that CAD systems should "be thought of not as supporting an individual engineer, but as supporting engineering work." The nature of CAD systems as visual mechanisms for sharing design information echoes the

findings of this research regarding establishment of shared understanding of design using a system representation.

8.6 Recommendations for Further Research

This study has pointed the way to several promising areas for further research. Examination of programs in other defense sectors such as airframes or munitions in which the impact of rapidly changing technology is less significant could lead to a greater understanding of the relationship between technology changes and adaptation of programs during the design phase. Study of larger programs could also broaden insight by investigating how well the findings identified in this research are impacted by scaling considerations such as the number of organizations, individuals and tasks associated with a program.

Another compelling area for further research is the study of different types of system representations. Lower fidelity SRs such as drawings or mockups require fewer resources while providing a reduced exchange of knowledge between stakeholders. Studying a wide variety of SRs might provide insight into the best value approach for programs facing different mixtures of uncertainty and resource constraints. Research could also be done to compare the use of a SR to that of analytical calculations to determine which emphasis areas might best be supported through these two means.

Interoperability with other systems surfaced as a problematic consideration in several of the cases. Programs found it difficult to capture details of external systems since they were typically controlled by other organizations. Also, these systems were

often on a different developmental timeline. Further research into ways of effectively portraying inter-system designs in conjunction with the use of a system representation could help fill a current need in an era in which interoperability is often a key determinant of system value to the warfighter.

Time considerations such as “time boxing” (Highsmith, 2000) may play a larger role in the contractor design processes than was apparent in this research. The focus applied in this effort on observing collaborative practices may have obscured the significance of some time-related internal contractor practices. The implications for adaptability of “time boxing” as part of contractor design processes (both internal and collaborative) could be a fertile subject for further investigation.

8.7 Final Thoughts

This research has contributed recommendations for policy makers and Air Force acquisition stakeholders concerning adaptability during the design phase of acquisition. Insight into the benefits of using a system representation in combination with specific stakeholder roles has the potential to help future Air Force programs be more adaptable in an era of considerable uncertainty and rapid change.

From a theoretical perspective, CAS theory in an organizational context is still in its infancy. The findings that emerged from this research provide an indication that CAS principles are important considerations to understand inter-organizational interactions and adaptability in enterprises involving multiple stakeholders.

This study expands on the intra-organizational application of boundary objects addressed in previous research (e.g. Carlile, 1997 and 2002; Bernstein 2000) to address a

situation in which multiple organizations must collaborate to share knowledge and reach a common understanding of distributed information across organizational boundaries.

The research also clarifies the importance of adaptability in the design phase for Air Force acquisition. This importance stems from two co-existent factors. First - the definition of value for the warfighter changes and emerges over time. Second – limited resources are available to continually modify fielded weapon systems. The design phase provides a unique opportunity for stakeholders to visualize and interact with a system when changes are still economically feasible and when user understanding of how the system will be operated can be brought to bear. Providing the most capability within established financial constraints through the ability to adapt represents delivery of best value to the Air Force's customer – the operational warfighter.

List of References

Author's note: the following references were either directly cited in the dissertation or were of benefit in formulating the ideas and approaches used in this research.

Aerospace Systems

Boppe, C. (Summer, 2000). Class lecture notes, ESD.33J, *Systems Engineering*, Massachusetts Institute of Technology.

Crawley, E. (Fall, 2000). Class lecture notes, 16.882, *System Architecture*, Massachusetts Institute of Technology.

DeNeufville, R. (1990). Applied Systems Analysis: Engineering Planning and Technical Management. New York, McGraw-Hill, Inc.

Hughes, T. P. (1998). Rescuing Prometheus. New York, Vintage Books.

Hughes, A. C. a. H., Thomas P., Ed. (2000). Systems, Experts, and Computers. Cambridge, MA, MIT Press.

Mindel, D. (Fall, 2001). Class lecture notes, ESD.83, *Seminar in Engineering Systems*, Massachusetts Institute of Technology.

Rechtin, E. Systems Architecting of Organizations. (2000.) Boca Raton, CRC Press.

Steward, Don (1995.) Systems Analysis and Management: Structure, Strategy and Design. Brookline, MA, Problematics.

Lean Principles

Lean Aerospace Initiative (LAI) (1998.) "Lean Enterprise Model."

Murman, et al. (2002). Lean Enterprise Value: Insights from MIT's Lean Aerospace Initiative. New York, Palgrave.

Nightingale, D. (Fall, 2000.) class lecture notes, 16.852, *Integrating the Lean Enterprise*, Massachusetts Institute of Technology.

Stanke, Alexis (2001.) "A Framework for Achieving Lifecycle Value in Product Development." Massachusetts Institute of Technology, Unpublished Masters Thesis.

Womack, James P. and Jones, Daniel T. (1996.) Lean Thinking: Banish Waste and Create Wealth in Your Corporation. New York, Simon and Schuster.

Complex Adaptive Systems Theory

Arthur, W.B., Durlauf, S.N. and D.A. Lane, eds. (1997). The Economy as an Evolving Complex System II, Proceedings Volume XXVII, Santa Fe Studies in the Sciences of Complexity. Reading, MA: Addison-Wesley.

Axelrod, R. (1997). The Complexity of Cooperation. Princeton, NJ: Princeton University Press.

Gell-Mann, Murray (1994). The Quark and the Jaguar: Adventures in the Simple and the Complex. New York, W. H. Freeman and Company.

Highsmith, James (2000.) Adaptive Software Development. New York, Dorset House.

Holland, John (1995). Hidden Order: How Adaptation Builds Complexity. Reading, MA, Addison-Wesley Publishing Company.

Kauffman, Stuart (1995). At Home in the Universe. New York, Oxford University Press.

Pascale, Richard T. (Spring, 1999.) "Surfing the Edge of Chaos." Sloan Management Review.

Prigogine, I. (1984). Order Out of Chaos. New York, Bantam Books.

Stacey, Ralph (1996). Complexity and Creativity in Organizations. San Francisco, Berrett-Koehler Publishers.

Waldrop, M. Mitchell (1992.) Complexity: the Emerging Science at the Edge of Order and Chaos. New York, Touchstone.

Wheatley, M. (1999.) Leadership and the New Science: Discovering Order in a Chaotic World. 2nd edition. San Francisco, Berrett-Koehler.

Boundary Objects

Bernstein, Joshua (2001.) "Multidisciplinary Design Problem Solving on Product Development Teams." Massachusetts Institute of Technology, Unpublished Doctoral Dissertation.

Carlile, Paul Reuben (1997.) "Understanding Knowledge Transformation in Product Development: Making Knowledge Manifest through Boundary Objects." University of Michigan, Unpublished Doctoral Dissertation.

Carlile, Paul Reuben (2002.) "A Pragmatic View of Knowledge and Boundaries: Boundary Objects in New Product Development." *Organization Science.*

Robertson, David Chandler. (1990.) "CAD Systems and Communication in Design Engineering: A Test of the Information Processing Model." Massachusetts Institute of Technology, Unpublished Doctoral Dissertation.

Star, S. L. (1989.) "The structure of ill-structured solutions: Boundary Objects and Heterogeneous Distributed Problem Solving." M. Huhns and L. Gasser, eds. *Readings in Distributed Artificial Intelligence.* Menlo Park, CA, Morgan Kaufman,

Defense Reform Literature

"A Quest for Excellence: Final Report to the President." (1986). President's Blue Ribbon Commission on Defense Management.

Adams, G. (1983). Controlling Weapons Costs: Can the Pentagon Reforms Work? New York, Council on Economic Priorities.

Farrell, T. (1997). Weapons Without a Cause: the Politics of Weapons Acquisition in the United States. New York, St. Martin's Press.

Fox, J. (1988). The Defense Management Challenge: Weapons Acquisition. Boston, Harvard Business School Press.

Gregory, W. The Defense Procurement Mess. Lexington, MA, Lexington Books.

Lebovic, J. (1996). Foregone Conclusions: U.S. Weapons Acquisition in the Post-Cold War Transition. Boulder, CO, Westview Press.

"New Thinking and American Defense Technology." (1990) Carnegie Commission on Science, Technology, and Government, first edition.

"New Thinking and American Defense Technology." (1993) Carnegie Commission on Science, Technology, and Government, second edition.

Peck, M and F. Scherer (1962). The Weapons Acquisition Process: An Economic Analysis. Boston, Division of Research, Graduate School of Business Adminstration, Harvard University.

Defense Acquisition Policy and Regulations

Air Force Chief of Staff and Secretary Memo (11 March 2002.) "Agile Acquisition and Logistics Transformation Imperatives."

Air Force Instruction 63-101 (1994.) "Acquisition System."

Air Force Policy Directive 10-9 (2000.) "Lead Operating Command Weapon System Management."

Assistant Secretary of the Air Force (Acquisition) Memo (4 June 2002.) "Reality-Based Acquisition System Policy for All Programs."

Department of Defense MIL-STD-1521B (1986.) "Technical Reviews and Audits for Systems, Equipment, and Computer Software."

Department of Defense Publication (2002.) "Interim Defense Acquisition Guidebook."

Draft Air Force Instruction 63-101 (2003.) "Acquisition System."

HQ Air Force Policy Memo (23 June 1997.) "Open Communication with Industry."

Joint Staff Memo (7 October 2002.) "Changes to the Requirements Generation System."

Under Secretary of Defense (Acquisition, Technology and Logistics) Memo (12 April 2002.) "Evolutionary Acquisition and Spiral Development."

Organizational Behavior

Allen, T. (1984.) Managing the Flow of Technology. Paperback edition. Cambridge, MIT Press.

Robbins, S. P. (1998.) Organizational Behavior. Upper Saddle River, NJ, Prentice-Hall International, Inc.

Schrage, Michael. (1995.) No More Teams: Mastering the Dynamics of Creative Collaboration. New York, Doubleday.

Wilson, J. (1989.) Bureaucracy: What Government Agencies Do and Why They Do It. USA, Basic Books.

Social Science Research

Krathwohl, D. (1998.) Methods of Educational and Social Science Research: an Integrated Approach. Second edition. New York, Addison Wesley.

Rebentisch, E. (1999.) Class lecture notes, ESD.843J, *Seminar in Social Science Research Methods*. (ESD.843J), Massachusetts Institute of Technology.

Appendix A: Questionnaire

Questionnaire on Stakeholder Collaboration
During the Design Phase of Air Force Acquisition

Several system program offices at Electronic Systems Center (ESC), along with their user and contractor counterparts, will be asked to participate in case studies to understand mechanisms for collaboration during the design phase of an acquisition program. Collaboration mechanisms called “system representations” (such as prototypes, beta software releases, etc.) provide a focus for sharing design information and other system-related knowledge between stakeholders (user, SPO, and contractor). The purpose of this research is to better understand the role of system representations in the sharing of stakeholder knowledge. Knowledge sharing may facilitate identification and evaluation of requirements and design decisions while program adaptations are still affordable.

The research group conducting this work is the Lean Aerospace Initiative (LAI) at MIT. LAI is a consortium of academia, industry and government organizations dedicated to improving business processes and practices in the aerospace industry.

We would like to emphasize that participation in this research is completely voluntary. You are free to refuse to answer any question you are either uncomfortable with or uncertain about, or to withdraw your participation at any time. We understand that you may have concerns about confidentiality. Several measures will be taken to ensure that your responses will remain confidential. Data will be processed only by MIT researchers.

All analysis of the data will be represented in non-attributable or aggregated form. Excerpts from the questionnaire or the follow-on interview may be made part of the research results, but under no circumstances will your name or any identifying characteristics be included. Furthermore, no individual program will be identified in the analysis or reporting of the responses. We understand that the success of any research depends upon the quality of the information on which it is based, and we take seriously our responsibility to ensure that any information you entrust to us will be protected.

If you have any questions about this effort, please feel free to contact:

Lt. Col. Robert Dare, USAF
Lean Aerospace Initiative
Massachusetts Institute of Technology
Email: darer@mit.edu Phone: (617) 258-7984 or (617) 332-3030

SPECIAL INSTRUCTIONS

Please complete the questionnaire as completely as possible before the interview, and note any issues you may have with particular questions. I will collect your questionnaire and go over your responses with you in the interview, so you will have an opportunity to clarify or expand on your answers, or ask any questions about the research.

Thank you for lending your time to this effort!

PHASE 1: QUESTIONNAIRE (please complete in advance of interview)

Personal and program information (please fill in any blanks and make any necessary corrections):

Name:
Phone #:
Email:
Organization:
Program Name:

Stakeholder Group: __ User __ SPO __ Prime contractor

Important! The following definition is central to completing the questionnaire. Please read through the definition of system representation carefully before continuing.

System Representation (SR). *A system representation is a mechanism for capturing information about a system for the specific purpose of sharing that information between user, acquirer and contractor communities during the design phase. Examples of a SR include prototypes, software beta releases, mockups, drawings, etc. The SR itself, along with a supporting collaborative pattern of interaction between stakeholders, captures design information and additional system-related information (stakeholder knowledge) such as dynamic (changing) user needs, cost and schedule data and/or system constraints.*

Based on discussions with SPO program management, the following SR was used on this program during the design phase:

Program System Representation: ____
____

*****All future references to a System Representation (SR) refer explicitly to this SR*****

Please answer the following questions to the best of your ability.

You may find that you do not have information or insight into some questions. If you have issues or observations because of the questions, please make a note. We will discuss these aspects during the interview.

1. Your experience with this acquisition program:

a. What responsibilities did/do you have for this project? (e.g. program manager, MAJCOM Program Element Monitor (PEM), Wing representative, etc.)

____

b. During what time period have you worked in this capacity on this program?

From: ______________ to _______________

2. Design phase timeline.

The SPO has provided the following information to delineate the design phase for this program. These dates serve to bracket the period of interest of this case study.

a. What was the <u>start date of the "design phase"</u> for this program?

b. What is the planned or actual <u>completion date</u> of the <u>Critical Design Review</u> (CDR)? (If multiple CDRs took place or the meeting(s) spanned several days, list the last date on which a formal CDR session was held).

3. Program context - uncertainty at the start of the design phase

a. Technical requirements derived from user needs make up a program's requirements baseline. Which of the following factors were major contributors to uncertainty in the <u>requirements baseline</u> at the start of the design phase? *Check all that apply:*

__ Anticipated program funding level	__ Technology maturity
__ Threat change	__ Other ___________________________
__ Rapid technology change	__ Other ___________________________
__ Diverse user perspectives	__ Other ___________________________

b. Given the presence of these factors, what was the level of uncertainty in your program's <u>requirements baseline</u> at the start of the design phase?

__ Low (extremely stable requirements baseline)
__ Medium-Low (fairly stable requirements baseline)
__ Medium (changes in requirements baseline likely)
__ Medium High (many changes in requirements baseline likely)
__ High (strong probability of many changes in the requirements baseline)

c. What was the level of uncertainty in your program's user concept of operations (definition of how the system is intended to be used in an operational setting) at the start of the design phase?

__ Low (extremely well defined operations concept)
__ Medium-Low (fairly well defined operations concept)
__ Medium (some definition of operations concept)
__ Medium High (minimal definition of operations concept)
__ High (no definition of operations concept)

4. Timing and iteration of system representation (SR):

a. Creating a SR permits multiple stakeholders to evaluate a system during its design phase, with the SR serving as a common frame of reference. This study assumes the contractor generates the SR. On what date was the SR made available for review to user and/or SPO representatives? (Information provided by SPO):

User: ____ SPO: ____

b. How many times (distinct iterations) were different versions of the SR used to collect feedback from the user and/or SPO communities? For example, if two beta releases of software or two versions of a prototype were generated and shared with stakeholders to collect feedback before the design was finalized, this would count as two iterations.

(Circle your	User:	Not sure	0	1	2	3	4	5	6 or more
choice)	SPO:	Not sure	0	1	2	3	4	5	6 or more

5. Stakeholder participation with system representation (SR)

a. Which user representatives (MAJCOM HQ and/or Wing level) were involved in SR creation, review or feedback activity (name, organization)?

Name (Optional)	Organization (Office Symbol)

b. What roles did these user representatives have (collectively) in the context of the SR? *(Check all that apply).*

__ Provide input to SR creation or revision
__ Evaluate the SR to generate comments
__ Design, build, and/or code the SR
__ Implement changes to the SR
__ Coordinate content and feedback for SR development/modification
__ Other ____
__ Other ____

c. Which contractor representatives were involved in SR creation, review or feedback activity (name, functional/team affiliation)?

Name (Optional)	Functional/Team Affiliation

d. What roles did these contractor representatives have (collectively) in the context of the SR? *(Check all that apply).*

__ Provide input to SR creation or revision
__ Evaluate the SR to generate comments
__ Design, build, and/or code the SR
__ Implement changes to the SR
__ Coordinate content and feedback for SR development/modification
__ Other ____
__ Other ____

e. Which SPO representatives were involved in SR creation, review or feedback activity? (name, title)

Name (Optional)	Title/Responsibility

f. What roles did these SPO representatives have (collectively) in the context of the SR? *(Check all that apply).*

__ Provide input to SR creation or revision
__ Evaluate the SR to generate comments
__ Design, build, and/or code the SR
__ Implement changes to the SR
__ Coordinate content and feedback for SR development/modification
__ Other ____
__ Other ____

6. Knowledge sharing and system representation (SR) creation/revision

Both initial SR creation and iterative changes to the SR (if applicable) involve collection of knowledge related to the system that is being designed. This knowledge may be distributed among stakeholders. The following questions explore the extent to which different stakeholders contributed their knowledge to SR creation and iteration.

a. To what degree were stakeholders (user, SPO and contractor) involved in contributing knowledge during the original SR creation process? *If you have no firsthand insight for a particular stakeholder, leave that section blank. Otherwise, circle one answer for each stakeholder below.*

	Not involved	Minimally involved	Moderately involved	Very involved	Extensively involved
User	1	2	3	4	5
SPO	1	2	3	4	5
Contractor	1	2	3	4	5

b. To what degree were stakeholders (user, SPO and contractor) involved in contributing knowledge during revisions of the SR (if applicable)? *If you have no firsthand insight for a particular stakeholder, leave that section blank. Otherwise, circle one answer for each stakeholder below.*

	Not involved	Minimally involved	Moderately involved	Very involved	Extensively involved
User	1	2	3	4	5
SPO	1	2	3	4	5
Contractor	1	2	3	4	5

c. To what degree did the user, SPO and contractor contribute their knowledge to initial SR creation and/or SR iteration in the following knowledge areas? Answer for the total (aggregate) of each stakeholder's contribution during the design phase.

> ***Instructions***:
> - *If you have no firsthand insight for a particular knowledge area and stakeholder, leave that section blank.*
> - *Circle one of the following answers for each of the three stakeholders:*
>   *NA – not applicable, stakeholder did not contribute knowledge in this area*
>   *S – stakeholder contributed a small amount of knowledge in this area to the SR*
>   *M – stakeholder contributed a moderate amount of knowledge in this area to the SR*
>   *L – stakeholder contributed a large amount of knowledge in this area to the SR*
> - *Please add additional knowledge areas that you believe were shared in the spaces marked "other."*

Knowledge of:	Contributed By User				Contributed By SPO				Contributed By Contractor			
Evolving user needs (beyond current requirements baseline)	NA	S	M	L	NA	S	M	L	NA	S	M	L
User operational considerations	NA	S	M	L	NA	S	M	L	NA	S	M	L
User maintenance considerations	NA	S	M	L	NA	S	M	L	NA	S	M	L
Performance levels of potential design choices	NA	S	M	L	NA	S	M	L	NA	S	M	L
Cost implications of potential design choices	NA	S	M	L	NA	S	M	L	NA	S	M	L
Schedule implications of potential design choices	NA	S	M	L	NA	S	M	L	NA	S	M	L
Design detail (specifics of form, fit and/or function)	NA	S	M	L	NA	S	M	L	NA	S	M	L
Program cost constraints	NA	S	M	L	NA	S	M	L	NA	S	M	L
Program schedule constraints	NA	S	M	L	NA	S	M	L	NA	S	M	L
Interoperability considerations	NA	S	M	L	NA	S	M	L	NA	S	M	L
Upgradeability considerations	NA	S	M	L	NA	S	M	L	NA	S	M	L
Sustainment considerations (e.g. documentation, training, sparing)	NA	S	M	L	NA	S	M	L	NA	S	M	L
Other:	NA	S	M	L	NA	S	M	L	NA	S	M	L
Other:	NA	S	M	L	NA	S	M	L	NA	S	M	L
Other:	NA	S	M	L	NA	S	M	L	NA	S	M	L

7. Collaborative communication and shared system understanding.

The SR may act as a focus for building shared understanding of the system that is being designed. The following questions address communication between stakeholders before and after the SR is available for community review.

a. Before initial SR release, how frequently did you use the following communication mechanisms to exchange information about the system with the other two stakeholders? *Use both response areas – one for communications with SPO/user/contractor, and the next for communications with SPO/user/contractor.*

Comm. with …	Used less than monthly	Used monthly	Used several times per month	Used weekly	Used several times per week	Used daily	Used several times per day
Email	1	2	3	4	5	6	7
Telephone (one-on-one)	1	2	3	4	5	6	7
Group telephone calls/VTCs	1	2	3	4	5	6	7
Meetings	1	2	3	4	5	6	7
Documentation exchange	1	2	3	4	5	6	7
Other ______________________	1	2	3	4	5	6	7
Other ______________________	1	2	3	4	5	6	7

Comm. with …	Used less than monthly	Used monthly	Used several times per month	Used weekly	Used several times per week	Used daily	Used several times per day
Email	1	2	3	4	5	6	7
Telephone (one-on-one)	1	2	3	4	5	6	7
Group telephone calls/VTCs	1	2	3	4	5	6	7
Meetings	1	2	3	4	5	6	7
Documentation exchange	1	2	3	4	5	6	7
Other ______________________	1	2	3	4	5	6	7
Other ______________________	1	2	3	4	5	6	7

b. After initial SR release, how frequently did you use the following communication mechanisms to exchange information about the system or the SR with the other two stakeholders? *Use both response areas – one for communications with SPO/user/ contractor, and the next for communications with SPO/user/contractor.*

Comm. with …	Used less than monthly	Used monthly	Used several times per month	Used weekly	Used several times per week	Used daily	Used several times per day
Email	1	2	3	4	5	6	7
Telephone (one-on-one)	1	2	3	4	5	6	7
Group telephone calls/VTCs	1	2	3	4	5	6	7
Meetings	1	2	3	4	5	6	7
Documentation exchange	1	2	3	4	5	6	7
Other ______________________	1	2	3	4	5	6	7
Other ______________________	1	2	3	4	5	6	7

(7b, continued)

Comm. with …	Used less than monthly	Used monthly	Used several times per month	Used weekly	Used several times per week	Used daily	Used several times per day
Email	1	2	3	4	5	6	7
Telephone (one-on-one)	1	2	3	4	5	6	7
Group telephone calls/VTCs	1	2	3	4	5	6	7
Meetings	1	2	3	4	5	6	7
Documentation exchange	1	2	3	4	5	6	7
Other ______________________	1	2	3	4	5	6	7
Other ______________________	1	2	3	4	5	6	7

c. How valuable were these communication mechanisms in sharing knowledge about the system or the SR between stakeholders (user, SPO, and contractor)? How valuable were any other mechanisms (identified in the previous question)?

	Not Valuable		Moderately Valuable		Exceptionally Valuable
Email	1	2	3	4	5
Telephone (one-on-one)	1	2	3	4	5
Group telephone calls/VTCs	1	2	3	4	5
Meetings	1	2	3	4	5
Documentation exchange	1	2	3	4	5
Other ___________________________	1	2	3	4	5
Other ___________________________	1	2	3	4	5

d. How long did it typically (on average) take to get feedback from another stakeholder in response to an information request about the system being designed or the SR?

	Not Observed	One month or more	Two weeks to a month	One to two weeks	One day to one week	Less than one day
From SPO	N/O	1	2	3	4	5
From Contractor	N/O	1	2	3	4	5
From User	N/O	1	2	3	4	5

e. What would be a <u>desired length of time</u> (on average) to get feedback from another stakeholder in response to an information request about the system being designed or the SR?

	Not Observed	One month or more	Two weeks to a month	One to two weeks	One day to one week	Less than one day
From SPO	N/O	1	2	3	4	5
From Contractor	N/O	1	2	3	4	5
From User	N/O	1	2	3	4	5

8. Stakeholder involvement in decision-making

a. To what degree were stakeholders (user, SPO and contractor) involved in the decision-making process to approve elements of the original SR?

	Not involved	Minimally involved	Moderately involved	Very involved	Approval/veto authority
User	1	2	3	4	5
SPO	1	2	3	4	5
Contractor	1	2	3	4	5

b. To what degree were stakeholders (user, SPO and contractor) involved in the decision-making process to approve SR changes? If the SR was not iteratively changed, leave this question blank.

	Not involved	Minimally involved	Moderately involved	Very involved	Approval/veto authority
User	1	2	3	4	5
SPO	1	2	3	4	5
Contractor	1	2	3	4	5

c. To what degree were stakeholders (user, SPO and contractor) involved in the decision-making process to approve system-level requirements changes (Class I Engineering Change Proposals)?

	Not involved	Minimally involved	Moderately involved	Very involved	Approval/veto authority
User	1	2	3	4	5
SPO	1	2	3	4	5
Contractor	1	2	3	4	5

d. To what degree were stakeholders (user, SPO and contractor) involved in the decision-making process to approve system design changes (modifications to stated contractor design details)?

	Not involved	Minimally involved	Moderately involved	Very involved	Approval/veto authority
User	1	2	3	4	5
SPO	1	2	3	4	5
Contractor	1	2	3	4	5

9. Knowledge transfer factors. Factors related to organizational culture, processes/procedures, and tools can influence knowledge transfer between stakeholders during the design phase of an acquisition program. The following questions ask you to rate (and contribute your own) factors that have either assisted **or** inhibited transfer of knowledge between stakeholders on your program.

a. For your program, how much have the following elements of organizational culture **assisted** knowledge transfer with other stakeholders regarding the system being designed? *Please add any additional elements of organizational culture of particular significance on your program in the spaces labeled "other" below.*

	Not Used	No Effect	Slight Effect	Moderate Effect	Strong Effect	Enormous Effect
Leadership emphasis on collaboration	N/A	1	2	3	4	5
Clear, shared definition of stakeholder responsibilities	N/A	1	2	3	4	5
Shared sense of mission	N/A	1	2	3	4	5
Shared goals (e.g. cost, performance targets, delivery schedules, etc)	N/A	1	2	3	4	5
Environment of trust	N/A	1	2	3	4	5
Contractor future business or technology needs	N/A	1	2	3	4	5
Other ______________________________		1	2	3	4	5
Other ______________________________		1	2	3	4	5
Other ______________________________		1	2	3	4	5

b. For your program, how much have the following processes or practices **assisted** knowledge transfer with other stakeholders regarding the system being designed? *Please add any additional processes or practices of particular significance on your program in the spaces labeled "other" below.*

	Not Used	No Effect	Slight Effect	Moderate Effect	Strong Effect	Enormous Effect
Meetings/working groups	N/A	1	2	3	4	5
Joint configuration control boards	N/A	1	2	3	4	5
Continuity of user representatives (same personnel over time)	N/A	1	2	3	4	5
Empowered user representatives (able to speak and decide for user organization)	N/A	1	2	3	4	5
Use of liaisons at other stakeholder sites	N/A	1	2	3	4	5
Collocation of stakeholders	N/A	1	2	3	4	5
Collaboration on previous efforts	N/A	1	2	3	4	5
Use of schedule deadlines to drive responsiveness	N/A	1	2	3	4	5
Direct interaction of technical personnel	N/A	1	2	3	4	5
Other ______________________________		1	2	3	4	5
Other ______________________________		1	2	3	4	5
Other ______________________________		1	2	3	4	5

c. For your program, how much have the following tools **assisted** knowledge transfer with other stakeholders regarding the system being designed? *Please add any additional tools of particular significance on your program in the spaces labeled "other" below.*

	Not Used	No Effect	Slight Effect	Moderate Effect	Strong Effect	Enormous Effect
Groupware (collaborative software)	N/A	1	2	3	4	5
Shared databases	N/A	1	2	3	4	5
Market surveys	N/A	1	2	3	4	5
User surveys	N/A	1	2	3	4	5
Trade studies	N/A	1	2	3	4	5
Modeling tools	N/A	1	2	3	4	5
Simulation tools	N/A	1	2	3	4	5
Analytical tools	N/A	1	2	3	4	5
Cost models or estimates	N/A	1	2	3	4	5
Other ______________________________		1	2	3	4	5
Other ______________________________		1	2	3	4	5
Other ______________________________		1	2	3	4	5

d. For your program, how much have the following elements of organizational culture **inhibited** knowledge transfer with other stakeholders regarding the system being designed? *Please add any additional elements of organizational culture of particular significance on your program in the spaces labeled "other" below.*

	Not Used	No Effect	Slight Effect	Moderate Effect	Strong Effect	Enormous Effect
SPO lacks user perspective	N/A	1	2	3	4	5
User lacks SPO perspective	N/A	1	2	3	4	5
Contractor lacks user perspective	N/A	1	2	3	4	5
User lacks contractor perspective	N/A	1	2	3	4	5
Contractor lacks SPO perspective	N/A	1	2	3	4	5
SPO lacks contractor perspective	N/A	1	2	3	4	5
User had conflicting priorities/divided attention	N/A	1	2	3	4	5
Contractor had conflicting priorities/divided attention	N/A	1	2	3	4	5
SPO lacks incentive to collaborate	N/A	1	2	3	4	5
Contractor lacks incentive to collaborate	N/A	1	2	3	4	5
User lacks incentive to collaborate	N/A	1	2	3	4	5
Presence of personality conflicts	N/A	1	2	3	4	5
Other________________________________		1	2	3	4	5
Other________________________________		1	2	3	4	5
Other________________________________		1	2	3	4	5

e. For your program, how much have the following processes or practices **inhibited** knowledge transfer with other stakeholders regarding the system being designed? *Please add any additional processes or practices of particular significance on your program in the spaces labeled "other" below.*

	Not Used	No Effect	Slight Effect	Moderate Effect	Strong Effect	Enormous Effect
User engages late in design phase	N/A	1	2	3	4	5
Lack of SPO follow-up on identified issues	N/A	1	2	3	4	5
SPO understaffed	N/A	1	2	3	4	5
User understaffed	N/A	1	2	3	4	5
Contractor understaffed	N/A	1	2	3	4	5
SPO reluctance to communicate issues	N/A	1	2	3	4	5
User reluctance to communicate issues	N/A	1	2	3	4	5
Contractor reluctance to communicate issues	N/A	1	2	3	4	5
User under funded for collaborative involvement (e.g. travel dollars)	N/A	1	2	3	4	5
SPO under funded for collaborative involvement	N/A	1	2	3	4	5
Contractor under funded for collaborative involvement	N/A	1	2	3	4	5
Other________________________________		1	2	3	4	5
Other________________________________		1	2	3	4	5
Other________________________________		1	2	3	4	5

f. For your program, how much have the following tools **inhibited** knowledge transfer with other stakeholders regarding the system being designed? *Please add any additional tools of particular significance on your program in the spaces labeled "other" below.*

	Not Used	No Effect	Slight Effect	Moderate Effect	Strong Effect	Enormous Effect
Inadequate design tools	N/A	1	2	3	4	5
Inadequate cost estimating capability	N/A	1	2	3	4	5
Incompatible software	N/A	1	2	3	4	5
Other________________________________		1	2	3	4	5
Other________________________________		1	2	3	4	5
Other________________________________		1	2	3	4	5

THIS COMPLETES THE WRITTEN QUESTIONAIRRE.

THANKS FOR YOUR PARTICIPATION!

Appendix B: Analysis of Program Characteristics

Appendix B: Analysis of Program Characteristics

Certain program characteristics were identified during research design due to their potential significance as alternate explanations for levels of program adaptability. These program characteristics were requirements uncertainty (as of the start of the design phase), research and development (R&D) funding and duration of the design phase.

Requirements uncertainty was of interest because of the possibility that programs with high uncertainty might be predisposed to adapt more in the course of resolving ambiguities. Higher funding and longer duration were also potential explanations for greater levels of program adaptability. Still another possibility was that programs below a certain funding level might have an advantage due to smaller team sizes that would make them more agile. Exploration of these program characteristics aimed to discover if any of these possibilities had explanatory power regarding program adaptability levels.

B.1 Requirements Uncertainty

One of the questionnaire sections was designed to collect data on requirements uncertainty at the start of the design phase. However, the raw data (see Table B.1) from the questionnaire was not reliable. Stakeholder responses varied by 3 on the 5-point scale for two programs, and varied by 2 for another three programs. Only one of the eight programs had agreement between stakeholders, and just two programs varied by only one increment. This level of variation provided an indication that the responses were highly subjective.

Program	SPO Response	User Response	Contractor Response	Variance
A	Medium-low (2)	High (5)	Medium (3)	3
B	Medium-low (2)	Medium-low (2)	Medium-high (4)	2
C	Medium (3)	Medium-low (2)	Medium-high (4)	2
D	Medium-low (2)	Medium-high (4)	Medium (3)	2
E	Medium (3)	Medium (3)	Medium (3)	0
F	Medium-low (2)	High (5)	Medium-high (4)	3
G	Medium-low (2)	Medium (3)	Medium (3)	1
H	Medium-low (2)	Medium-low (2)	Medium (3)	1

Table B.1. Questionnaire data on initial requirements uncertainty

Since the questionnaire data was not conclusive, it was necessary to formulate a subjective impression of the requirements uncertainty for each program based on stakeholder comments in the interviews. Table B.2 includes an estimated uncertainty of low, medium or high based on the most relevant characterizations of the program provided in the stakeholder interviews.

Program	Interview comments	Uncertainty
A	User interested in exploring technology advances and new modeling algorithms	Medium
B	Existence of four user groups led to requirements clarification during the design phase	Medium
C	Requirements rigid due to budget constraints	Low
D	Detailed requirements not well understood	High
E	Requirements somewhat fluid depending on anomalies found during testing of another version of the software	Medium
F	Requirements well defined based on legacy system	Low
G	Functionality well defined based on legacy system; SPO stated that up-front emphasis on requirements definition had helped achieve on-time completion	Low
H	User not fully involved in requirements definition; major changes to requirements identified by user during design	High

Table B.2 Levels of program uncertainty

The estimated levels of requirements uncertainty provided a means of assessing whether this variable had a causal relationship with the level of program adaptability. Figure B.1 provides an ordered list of programs, based on adaptability level. The two most adaptable programs had medium levels of uncertainty. Programs H and D were the only programs with high requirements uncertainty, and they had, respectively, high and moderate levels of adaptability. This data shows a lack of correlation between requirements uncertainty and adaptability.

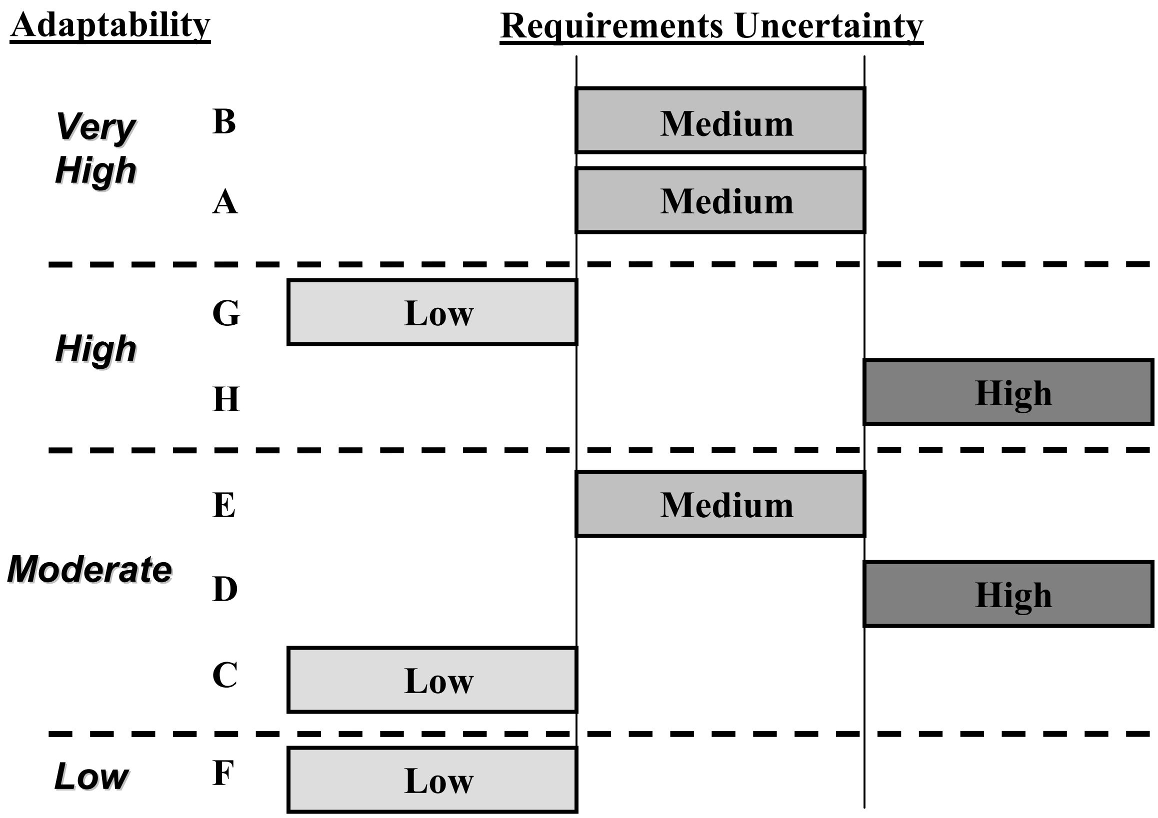


Figure B.1. Requirements uncertainty versus adaptability

B.2 Research and Development (R&D) Budget

R&D budgets varied from $13 million to $140 million. Figure B.2 shows the adaptability level and budget for the eight programs. Programs A ($24 million) and B ($23 million) demonstrate that a relatively small R&D budget did not inhibit adaptability. The lowest budget programs (E and C) and the highest budget programs (G and H) were not at either the high or the low end of adaptability performance. Based on this data, the budget level did not exert any causal influence on the adaptability of programs.

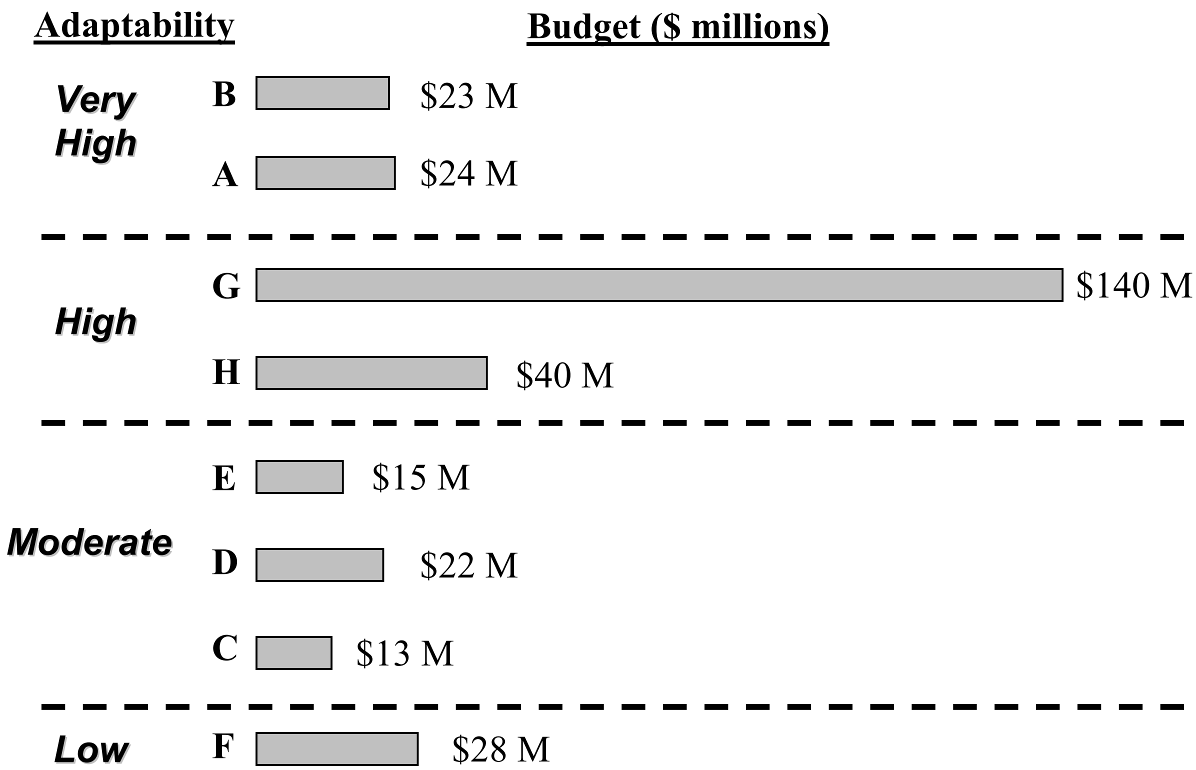


Figure B.2. R&D budget versus adaptability

B.3 Design Phase Duration

Figure B.3 shows the level of program adaptability and design phase durations. Design phases varied from 6 months to 43 months. It was clear from examining the 6 month program (Case E), which achieved moderate adaptability and implemented 12 collaborative changes, that the phenomena of stakeholder interactions, identification of potential collaborative changes and disposition of these changes was able to happen even

in this short timeframe. Three programs (C, D, and F) had more than twice the design time of Case E but had comparable or lower adaptability levels.

While there was a general trend for longer duration design phases to correlate to higher adaptability, programs B and E departed from that trend. Program B was in the highest tier of adaptability even though it's design phase was approximately one-third that of Program A. Program E achieved moderate adaptability with less than half the design time of the next quickest program. This data demonstrates that the design phase duration did not exert any causal influence on the adaptability of programs.

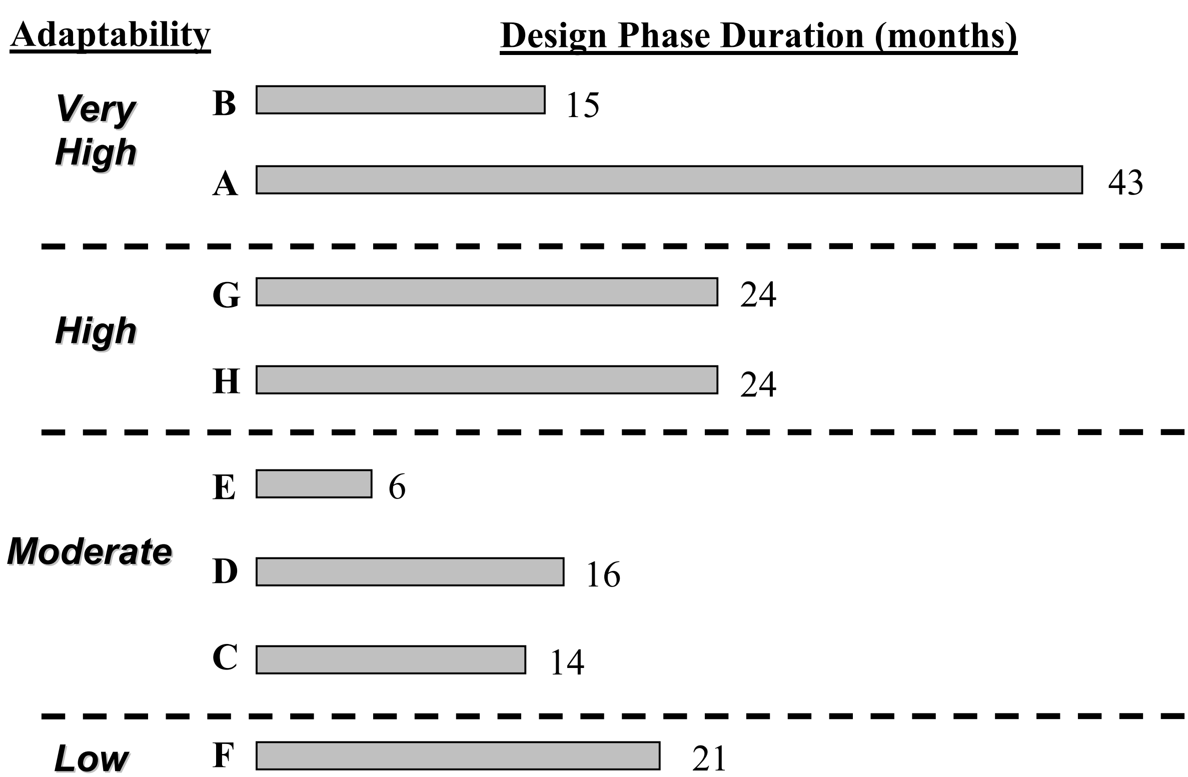


Figure B.3. Design phase duration versus adaptability

B.4 Summary of Program Characteristics

The three program characteristics that were evaluated were requirements uncertainty, funding and design phase duration. These characteristics did not show causal links to observed levels of program adaptability. This result is captured in the finding presented in Table B.3.

Finding #6: Requirements uncertainty, funding and design phase duration did not have a causal relationship with respect to levels of program adaptability.

Table B.3. Finding #6: program characteristics